

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 09-179883

(43)Date of publication of application : 11.07.1997

(51)Int.Cl.

G06F 17/30

G06F 12/00

G06F 12/00

(21)Application number : 08-044660

(71)Applicant : INTERNATL BUSINESS MACH
CORP <IBM>

(22)Date of filing : 01.03.1996

(72)Inventor : FUKUDA TSUYOSHI
MORIMOTO YASUHIKO
MORISHITA SHINICHI
TOKUYAMA TAKESHI

(30)Priority

Priority number : 07278690

Priority date : 26.10.1995

Priority country : JP

(54) METHOD AND DEVICE FOR DERIVING CONNECTION RULE BETWEEN DATA

(57)Abstract:

PROBLEM TO BE SOLVED: To make it possible to find the correlation between data which have a two-term numeral attribute and a true/false attribute.

SOLUTION: A plane is constituted with two numeral attributes first and divided into meshes, and data in the meshes (packet) and data which have true attributes are counted. This plane can be grasped as the plane image that the number of data corresponds to the gray level and the number of data having true attributes corresponds to the saturation. Then a permissible image which is an area that is convex to one axis of the plane is cut under specific conditions and a part where the correlation of data is strong is found. Then when the area as the cut permissible area meets conditions of a support maximization rule, etc., the area is shown to a user. Further, necessary attributes of data included in the area are extracted from a data base at need.

LEGAL STATUS

[Date of request for examination]

13.05.1998

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

3118181

[Date of registration]

06.10.2000

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-179883

(43) 公開日 平成9年(1997)7月11日

(51) Int.Cl. ⁸	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 17/30			G 0 6 F 15/403	3 4 0 Z
12/00	5 0 5		12/00	5 0 5
	5 2 0			5 2 0 A

審査請求 未請求 請求項の数23 O L (全 26 頁)

(21) 出願番号 特願平8-44660

(22) 出願日 平成8年(1996)3月1日

(31) 優先権主張番号 特願平7-278690

(32) 優先日 平7(1995)10月26日

(33) 優先権主張国 日本 (J P)

(71) 出願人 390009531

インターナショナル・ビジネス・マシーンズ・コーポレーション

INTERNATIONAL BUSINESS MACHINES CORPORATION

アメリカ合衆国10504、ニューヨーク州
アーモンク (番地なし)

(72) 発明者 福田 剛志

神奈川県大和市下鶴間1623番地14 日本アイ・ピー・エム株式会社 東京基礎研究所内

(74) 代理人 弁理士 合田 潔 (外2名)

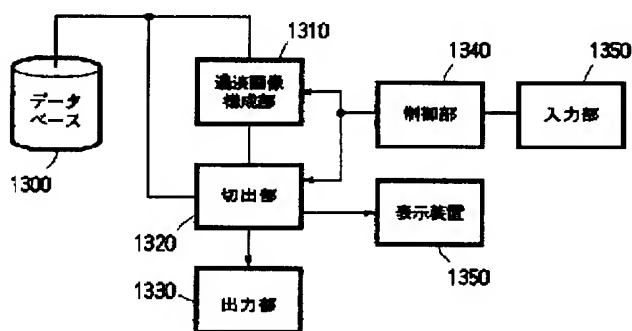
最終頁に続く

(54) 【発明の名称】 データ間結合ルール導出方法及び装置

(57) 【要約】

【課題】 2項の数値属性と真偽をとる属性を有するデータ間の相関を見出すための手法を提案すること。

【解決手段】 (1) 2つの数値属性により平面を構成し、この平面をメッシュ分割し、各メッシュ(バケットともいう)内のデータ数及び真偽をとる属性が真となったデータの数をカウントする。このような平面は、各メッシュをピクセルとした場合、データ数が濃淡度、真偽をとる属性が真となるデータの数が彩度に該当するような、平面画像として捉えることができる。(2) 所定の条件θに従い、平面の1つの軸に凸な領域である許容イメージを切り出し、データの相関の強い部分を見出す。(3) 切り出した許容イメージとなる領域が、サポート最大化ルール等の条件を満たしていれば、その領域をユーザに提示する。また、データベースからその領域に含まれるデータの必要な属性を引き出すことも、必要に応じて行う。



1

【特許請求の範囲】

【請求項1】2種類の数値属性と、1種類の真偽をとる属性を含むデータを有するデータベースにおいて、データ間の結合ルールを導き出す方法であって、前記2種類の数値属性に対応する2つの軸を有し且つ $N \times M$ 個のバケットに分割されている平面の各バケットに対応して、当該バケット（座標 (i, j) ）に属するデータの数 $u(i, j)$ 及び前記真偽をとる属性が真であるデータの数 $v(i, j)$ を記憶する平面構成ステップと、条件 θ を入力するステップと、

【数1】

$$\sum_{(i,j) \in S} g(i,j) = \sum_{(i,j) \in S} (v(i,j) - \theta u(i,j))$$

を最大にするような前記バケットの領域 S を前記平面から切り出す領域切出ステップと、切り出された前記領域 S 内に含まれるデータを出力するステップとを含むデータ間結合ルール導出方法。

【請求項2】入力された前記条件 θ とは異なる第2の条件 θ_2 を入力するステップと、

【数2】

$$\sum_{(i,j) \in S_2} g(i,j) = \sum_{(i,j) \in S_2} (v(i,j) - \theta_2 u(i,j))$$

を最大にするような前記バケットの第2の領域 S_2 を前記平面から切り出すステップと、

【数3】

$$\theta_3 = \frac{V(S_2) - V(S)}{U(S_2) - U(S)}$$

（前記領域 S_2 に含まれ且つ前記真偽をとる属性が真であるデータの数を $V(S_2)$ 、前記領域 S に含まれ且つ前記真偽をとる属性が真であるデータの数を $V(S)$ 、前記領域 S_2 に含まれるデータ数を $U(S_2)$ 、前記領域 S に含まれるデータ数を $U(S)$ とする。）を第3の条件として、

【数4】

$$\sum = \sum_{(i,j) \in S_3} (v(i,j) - \theta_3 u(i,j))$$

を最大にするような前記バケットの第3の領域 S_3 を前記平面から切り出すステップとをさらに含む請求項1記載のデータ間結合ルール導出方法。

【請求項3】前記切り出された領域 S 内の各バケットの $v(i, j) \neq u(i, j)$ が、前記平面全体のデータ数に対する前記平面全体の前記真偽をとる属性が真であるデータ数の割合に等しくなるよう $v(i, j)$ を変更するステップと、当該変更された $v(i, j)$ を用いて、入力された条件 θ_4 に従い、

【数5】

2

$$\sum_{(i,j) \in S_4} g(i,j) = \sum_{(i,j) \in S_4} (v(i,j) - \theta_4 u(i,j))$$

を最大にするような前記バケットの第4の領域 S_4 を切り出すステップとをさらに含む請求項1記載のデータ間結合ルール導出方法。

【請求項4】前記平面構成ステップが、複数の前記データから、 X 個のデータをランダムサンプリングするステップと、サンプリングされたデータを各前記数値属性についてソートし、 $X \cdot i \leq N$ ($i = 1, 2, \dots, N$) 番目に該当する数値及び $X \cdot n \leq M$ ($n = 1, 2, \dots, M$) 番目に該当する数値を記憶するステップと、記憶された前記数値を基準にして、前記複数のデータを $N \times M$ 個の前記バケットに入れるステップとを含む請求項1乃至3記載のデータ間結合ルール導出方法。

【請求項5】

【数6】

$$g(i, j) = v(i, j) - \theta u(i, j)$$

を評価値とし、

20 前記領域切出ステップが、前記平面の各列において、少なくとも1のバケットを含み且つ前記評価値が最大となるバケットの範囲を求める列内範囲導出ステップと、前記平面内の任意のバケット（座標 (m, t) ）の前列にあるもう1つのバケット（座標 $(m-1, 1)$ ）を含み且つ当該前列（第1列）までで最大の評価値を有する領域と、前記列内範囲導出ステップにより導出される、前記任意のバケット（座標 (m, t) ）及び前記もう1つのバケットと同一行であって前記任意のバケットと同一列にあるバケット（座標 $(m, 1)$ ）を含む評価値が最大となる範囲とを加えた領域を加算領域とする時、各前記任意のバケットに対し、それ自身及び当該任意のバケットを含む前記加算領域全体として評価値を最大化する前記もう1つのバケットを検出し、当該加算領域の評価値と共に記憶する検出記憶ステップと、前記平面全体で前記加算領域の評価値が最大となる前記任意のバケットを検出し、前記検出記憶ステップで検出された前記もう1つのバケットを用いて、領域 S を導き出すステップとを含む請求項1記載のデータ間結合ルール導出方法。

【請求項6】切り出されるべき領域に含まれる最低限のデータ数である最小サポート数 U_{min} を入力するステップと、前記切り出された領域 S に含まれるデータ数 $U(S)$ と前記最小サポート数 U_{min} と比較するステップと、前記比較の結果、 $U_{min} = U(S)$ であれば、当該領域 S を切り出されるべき領域として出力するステップと、前記比較の結果、 $U_{min} > U(S)$ 又は $U_{min} < U(S)$ の場合には、新たな条件 θ_5 にて、前記領域切出ステップを実施するステップとをさらに含む請求項1記載のデ

ータ間結合ルール導出方法。

【請求項7】切り出されるべき領域における前記真偽をとる属性が真であるデータの数の割合minconfを入力するステップと、

前記切り出された領域Sが、 $\text{minconf} \equiv V(S) / U(S)$ (S) ($U(S)$ は前記領域Sに含まれるデータ数、 $V(S)$ は前記領域Sに含まれ且つ前記真偽をとる属性が真であるデータの数)である場合には、当該領域Sを出*

$f(U(S), V(S))$

$$\begin{aligned} &= -V(S) \log \frac{V(S)}{U(S)} - (U(S) - V(S)) \log \frac{U(S) - V(S)}{U(S)} \\ &\quad - (V_{\text{sum}} - V(S)) \log \frac{V_{\text{sum}} - V(S)}{U_{\text{sum}} - U(S)} \\ &\quad - (U_{\text{sum}} - V_{\text{sum}} - U(S) + V(S)) \log \frac{U_{\text{sum}} - V_{\text{sum}} - U(S) + V(S)}{U_{\text{sum}} - U(S)} \end{aligned}$$

(U_{sum} は前記平面全体のデータ数、 V_{sum} は前記平面全体に含まれる前記真偽をとる属性が真のデータの数)を計算し、その値を前記領域Sに対応して記憶するエントロピー計算ステップと、
条件 θ を変更して前記領域切出ステップと前記エントロピー計算ステップを実行するステップと、

※

$$\begin{aligned} f(U(S), V(S)) &= U(S) \left(\frac{V(S)}{U(S)} - \frac{V_{\text{sum}}}{U_{\text{sum}}} \right)^2 \\ &\quad + (U_{\text{sum}} - U(S)) \left(\frac{V_{\text{sum}} - V(S)}{U_{\text{sum}} - U(S)} - \frac{V_{\text{sum}}}{U_{\text{sum}}} \right)^2 \end{aligned}$$

(U_{sum} は前記平面全体のデータ数、 V_{sum} は前記平面全体に含まれる前記真偽をとる属性が真のデータの数)を計算し、その値を前記領域に対応して記憶するインタクラスバリエーション計算ステップと、
条件 θ を変更して前記領域切出ステップと前記インタクラスバリエーション計算ステップとを実行するステップと、
 $f(U(S), V(S))$ を最大化する領域Sを出力するステップとをさらに含む請求項1記載のデータ間結合ルール導出方法。

【請求項10】2種類の数値属性と、1種類の真偽をとる属性を含むデータを有するデータベースにおいて、データ間の結合ルールを導き出す装置であって、前記2種類の数値属性に対応する2つの軸を有し且つ $N \times M$ 個のバケットに分割されている平面の各バケットに対応して、当該バケット(座標 (i, j))に属するデータの数 $u(i, j)$ 及び前記真偽をとる属性が真であるデータの数 $v(i, j)$ を記憶する平面構成手段と、
条件 θ を入力する入力手段と、

【数9】

$$\sum_{(i, j) \in S} g(i, j) = \sum_{(i, j) \in S} (v(i, j) - \theta u(i, j))$$

を最大にするような前記バケットの領域Sを前記平面か

※力するステップと、

$\text{minconf} < V(S) / U(S)$ 又は $\text{minconf} > V(S) / U(S)$ である場合には、新たな条件 θ_2 にて前記領域切出ステップを実施するステップとをさらに含む請求項1記載のデータ間結合ルール導出方法。

【請求項8】切り出された前記領域Sに対し、
【数7】

※ $f(U(S), V(S))$ を最大化する領域Sを出力するステップとをさらに含む請求項1記載のデータ間結合ルール導出方法。

20 【請求項9】切り出された前記領域Sに対し、
【数8】

ら切り出す領域切出手段と、

切り出された前記領域S内に含まれるデータを出力する手段と、
を有するデータ間結合ルール導出装置。

【請求項11】前記入力手段により、前記条件 θ とは異なる第2の条件 θ_2 を入力し、前記領域切出手段により、前記第2の条件 θ_2 に対応する第2の領域 S_2 を前記平面から切り出した場合に、

【数10】

$$\theta_3 = \frac{V(S_2) - V(S)}{U(S_2) - U(S)}$$

(前記領域 S_2 に含まれ且つ前記真偽をとる属性が真であるデータの数を $V(S_2)$ 、前記領域Sに含まれ且つ前記真偽をとる属性が真であるデータの数を $V(S)$ 、前記領域 S_2 に含まれるデータ数を $U(S_2)$ 、前記領域Sに含まれるデータ数を $U(S)$ とする。)を第3の条件として前記領域切出手段に出力する手段とをさらに有する請求項10記載のデータ間結合ルール導出装置。

【請求項12】前記切り出された領域S内の各バケットの $v(i, j) / u(i, j)$ が、前記平面全体のデータ数に対する前記平面全体の前記真偽をとる属性が真であるデータ

数の割合に等しくなるよう $v(i, j)$ を変更する手段と、当該変更された $v(i, j)$ 及び入力された条件 θ_4 でもって、前記領域切出手段が動作するように命令する手段とを有する請求項10記載のデータ間結合ルール導出装置。

【請求項13】前記平面構成手段が、複数の前記データから、 X 個のデータをランダムサンプリングする手段と、サンプリングされたデータを各前記数値属性についてソートし、 $X \cdot i / N$ ($i = 1, 2, \dots, N$) 番目に該当する数値及び $X \cdot n / M$ ($n = 1, 2, \dots, M$) 番目に該当する数値を記憶する手段と、記憶された前記数値を基準にして、前記複数のデータを $N \times M$ 個の前記バケットに入れる手段とを含む請求項10乃至12記載のデータ間結合ルール導出装置。

【請求項14】

【数11】

$$g(i, j) = v(i, j) - \theta_u(i, j)$$

を評価値とし、前記領域切出手段が、前記平面の各列において、少なくとも1のバケットを含み且つ前記評価値が最大となるバケットの範囲を求める列内範囲導出手段と、前記平面内の任意のバケット(座標(m, t))の前列にあるもう1つのバケット(座標($m-1, 1$))を含み且つ当該前列(第1列)までで最大の評価値を有する領域と、前記列内範囲導出手段により導出される、前記任意のバケット(座標(m, t))及び前記もう1つのバケットと同一行であって前記任意のバケットと同一列にあるバケット(座標($m, 1$))を含む評価値が最大となる範囲とを加えた領域を加算領域とする時、各前記任意のバケットに対し、それ自身及び当該任意のバケットを含む前記加算領域 $f(U(S), V(S))$

$$\begin{aligned} &= -V(S) \log \frac{V(S)}{U(S)} - (U(S) - V(S)) \log \frac{U(S) - V(S)}{U(S)} \\ &\quad - (V_{\text{sum}} - V(S)) \log \frac{V_{\text{sum}} - V(S)}{U_{\text{sum}} - U(S)} \\ &\quad - (U_{\text{sum}} - V_{\text{sum}} - U(S) + V(S)) \log \frac{U_{\text{sum}} - V_{\text{sum}} - U(S) + V(S)}{U_{\text{sum}} - U(S)} \end{aligned}$$

(U_{sum} は前記平面全体のデータ数、 V_{sum} は前記平面全体に含まれる前記真偽をとる属性が真のデータの数)を計算し、その値を前記領域 S に対応して記憶するエントロピ計算手段と、変更された条件 θ にて前記領域切出手段及び前記エントロピ計算手段が動作するように命ずる手段と、

* 域全体として評価値を最大化する前記もう1つのバケットを検出し、当該加算領域の評価値と共に記憶する検出記憶手段と、

前記平面全体で前記加算領域の評価値が最大となる前記任意のバケットを検出し、前記検出記憶手段により検出された前記もう1つのバケットを用いて、領域 S を導き出す手段とを含む請求項10記載のデータ間結合ルール導出装置。

【請求項15】切り出されるべき領域に含まれる最低限のデータ数である最小サポート数 U_{min} を入力する手段と、前記切り出された領域 S に含まれるデータ数 $U(S)$ と前記最小サポート数 U_{min} と比較する手段と、前記比較の結果、 $U_{\text{min}} \leq U(S)$ であれば、当該領域 S を切り出されるべき領域として出力する手段と、前記比較の結果、 $U_{\text{min}} > U(S)$ 又は $U_{\text{min}} < U(S)$ の場合には、新たな条件 θ_5 にて、前記領域切出手段が動作するように命ずる手段とを含む請求項10記載のデータ間結合ルール導出方法。

20 【請求項16】切り出されるべき領域における前記真偽をとる属性が真であるデータの数の割合 minconf を入力する手段と、

前記切り出された領域 S が、 $\text{minconf} \leq V(S) / U(S)$ ($U(S)$ は前記領域 S に含まれるデータ数、 $V(S)$ は前記領域 S に含まれ且つ前記真偽をとる属性が真であるデータの数)である場合には、当該領域 S を出力する手段と、

$\text{minconf} < V(S) / U(S)$ 又は $\text{minconf} > V(S) / U(S)$ である場合には、新たな条件 θ_6 にて前記領域切出手段が動作するように命ずる手段とを含む請求項10記載のデータ間結合ルール導出装置。

【請求項17】切り出された前記領域 S に対し、

【数12】

前記エントロピ計算手段に記憶された $f(U(S), V(S))$ を最大化する領域 S を出力する手段とを含む請求項10記載のデータ間結合ルール導出装置。

【請求項18】切り出された前記領域 S に対し、

【数13】

$$f(U(S), V(S)) = U(S) \left(\frac{V(S)}{U(S)} - \frac{V_{sum}}{U_{sum}} \right)^2 + (U_{sum} - U(S)) \left(\frac{V_{sum} - V(S)}{U_{sum} - U(S)} - \frac{V_{sum}}{U_{sum}} \right)^2$$

(U_{sum} は前記平面全体のデータ数、 V_{sum} は前記平面全体に含まれる前記真偽をとる属性が真のデータの数を計算し、その値を前記領域に対応して記憶するインタクラスバリエーション計算手段と、
変更された条件 θ にて前記領域切出手段と前記インタクラスバリエーション計算手段とを動作するように命ずる手段と、
前記インタクラスバリエーション計算手段に記憶された $f(U(S), V(S))$ を最大化する領域 S を出力する手段とを含む請求項10記載のデータ間結合ルール導出装置。

【請求項19】2種類の数値属性と、1種類の真偽をとる属性を含むデータを有するデータベースにおいて、コンピュータにデータ間の結合ルールを導き出させるプログラムコード手段を含む記憶装置であって、
コンピュータに、前記2種類の数値属性に対応する2つの軸を有し且つ $N \times M$ 個のバケットに分割されている平面の各バケットに対応して、当該バケット(座標 (i, j))に属するデータの数 $u(i, j)$ 及び前記真偽をとる属性が真であるデータの数 $v(i, j)$ を記憶させる平面構成プログラムコード手段と、
コンピュータに、条件 θ を入力させる入力プログラムコード手段と、
コンピュータに、

$$\sum_{(i,j) \in S} g(i,j) = \sum_{(i,j) \in S} (v(i,j) - \theta u(i,j))$$

を最大にするような前記バケットの領域 S を前記平面から切り出させる領域切出プログラムコード手段とを含む記憶装置。

【請求項20】コンピュータ及び前記入力プログラムコード手段により、前記条件 θ とは異なる第2の条件 θ_2 を入力し、コンピュータ及び前記領域切出プログラムコード手段により、前記第2の条件 θ_2 に対応する第2の領域 S_2 を前記平面から切り出した場合に、

【数15】

$$\theta_3 = \frac{V(S_2) - V(S)}{U(S_2) - U(S)}$$

(前記領域 S_2 に含まれ且つ前記真偽をとる属性が真であるデータの数を $V(S_2)$ 、前記領域 S に含まれ且つ前記真偽をとる属性が真であるデータの数を $V(S)$ 、前記領域 S_2 に含まれるデータ数を $U(S_2)$ 、前記領域

S に含まれるデータ数を $U(S)$ とする。)を第3の条件として前記領域切出プログラムコード手段に出力するプログラムコード手段とをさらに有する請求項19記載の記憶装置。

【請求項21】コンピュータに、前記切り出された領域 S 内の各バケットの $v(i, j)$ 及び $u(i, j)$ が、前記平面全体のデータ数に対する前記平面全体の前記真偽をとる属性のデータ数の割合に等しくなるよう $v(i, j)$ を変更させるプログラムコード手段と、
当該変更された $v(i, j)$ 及び入力された条件 θ を用いて、コンピュータと前記領域切出プログラムコード手段が動作するように命じるコンピュータプログラムコード手段とを有する請求項19記載の記憶装置。

【請求項22】前記平面構成プログラムコード手段が、コンピュータに、複数の前記データから、 X 個のデータをランダムサンプリングさせるプログラムコード手段と、
コンピュータに、サンプリングされたデータを各前記数値属性についてソートし、 $X \cdot i \div N$ ($i = 1, 2, \dots, N$) 番目に該当する数値及び $X \cdot n \div M$ ($n = 1, 2, \dots, M$) 番目に該当する数値を記憶させるプログラムコード手段と、

コンピュータに、記憶された前記数値を基準にして、前記複数のデータを $N \times M$ 個の前記バケットに入れさせるプログラムコード手段とを含む請求項19乃至22記載の記憶装置。

【請求項23】

【数16】

$$g(i, j) = v(i, j) - \theta u(i, j)$$

を評価値とし、

前記領域切出プログラムコード手段が、コンピュータに、前記平面の各列において、少なくとも1のバケットを含み且つ前記評価値が最大となるバケットの範囲を求めさせる列内範囲導出プログラムコード手段と、

前記平面内の任意のバケット(座標 (m, t))の前列にあるもう1つのバケット(座標 $(m-1, 1)$)を含み且つ当該前列(第1列)までで最大の評価値を有する領域と、前記列内範囲導出プログラムコード手段とコンピュータにより導出される、前記任意のバケット(座標 (m, t))及び前記もう1つのバケットと同一行であって前記任意のバケットと同一列にあるバケット(座標 $(m, 1)$)を含む評価値が最大となる範囲とを加えた領域を加算領域とする時、コンピュータに、各前記任意のバケットに対し、

それ自身及び当該任意のバケットを含む前記加算領域全体として評価値を最大化する前記もう1つのバケットを検出させ、当該加算領域の評価値と共に記憶させる検出記憶プログラムコード手段と、

コンピュータに、前記平面全体で前記加算領域の評価値が最大となる前記任意のバケットを検出させ、前記検出記憶プログラムコード手段で検出された前記もう1つのバケットを用いて、領域Sを導き出させるプログラムコード手段とを含む請求項19記載の記憶装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、データベースにおけるデータ相関の解析（データマイニングという。）に関し、より詳しくは2項の数値属性と1項の真偽をとる属性（真偽をとる条件又は0-1属性ともいう。）を有するデータ間の相関を見い出す手法に関する。

【0002】

【従来の技術】例えば、銀行の顧客を解析対象とし、流動性預金残高がいくらくらいで且つ年齢が何歳ぐらいの人であれば、定期預金残高が200万円以上になる人が全体の20%となるか、といった問題を実際に解くことを考える。この流通性預金残高及び年齢は、整数ではあるが連続数値であり、一方定期預金残高200万円以上というのは、200万円以上か未満かという分類になるので、真偽をとる属性を有するものである。真偽をとる属性は、例えば「顧客がクレジットカードを有しているか」や「顧客が男性であるか」といった問題と置き換えることも可能である。このような課題を解決することができれば、銀行はどのような人に、例えば新型の金融商品に関するダイレクトメールを送ればよいか簡単に分かるので、効率的な営業活動が行える。

【0003】従来、先に述べた真偽をとる属性間の相関を表現するルール（結合ルール、association rule）を高速に抽出するような研究は、データマイニングの分野において行われてきた。例えば、R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules between sets of items in large databases" In proceedings of the ACM SIGMOD Conference on Management of data, May 1993. や、R. Agrawal and R. Srikant, "Fast algorithms for mining association rules" In Proceedings of the 20th VLDB Conference, 1994. 等がある。

【0004】また、2項の数値データ間のルールを求める従来手法には、以下のようなものがある。

1. 強い線形相関を見い出すために、平面上の直線で、点集合を最適近似するものを探す方法。例えば、最小自乗法、再帰中央法等である。これら方法の欠点は、線形相関しか分からず、しかも相関係数の絶対値が0.5以下の場合に線形相関を用いて各データを予測すると精度が低く、現実にはほとんど役に立たない点にある。

2. 弱い大域相関を見い出すためには、2次元平面上で

正方形、長方形、又は円、楕円で面積に対して多くのデータを含むものを見い出す方法。例えば、計算幾何学アルゴリズムを利用するものである。この場合、計算時間が大きくなってしまいうという欠点がある。例えば円の場合、 $O(M^3)$ 以上の手間が掛かり得る（ $O(M)$ は、オーダー M の計算時間がかかることを示す。 M はデータ数である。）。また、取り出す相関領域としては決まった形をしたものしか扱うことができない。現実には、決まった形で適切にカバーできる場合は少ない。

3. 平面を正方メッシュに分割しておき、たくさんのデータを含むピクセルを取り出す方法。しかし、取り出されたピクセルの集合は連結でなく、バラバラなことが多いので、ルールとして見い出すのは困難である。

【0005】このような手法を用いると、上記の欠点の他に、データ間の多くのルールのうちで、意味のあるものと無意味なものとの区別が難しいという欠点もある。通常、相関に実用上の意味があるかどうかは人間の判断によらないといけなことが多く、1. や2. では特殊な相関しか取り出せないのも意味ある相関を見逃しやすく、3では出力を人間が見てルールを見い出せない。

【0006】以上のように、従来2項以上の数値属性を有するデータ間の相関を見い出す手法に有効なものはなく、真偽をとる属性間のデータマイニングの手法を組み合わせることができるわけでもない。よって、最初に述べた例のような、2項以上の数値属性と真偽をとる属性を有するデータ間の相関を見い出すために用いることができる手法は現在のところ提案されていない。

【0007】

【発明が解決しようとする課題】本発明は、以上のような点に鑑み、2項以上の数値属性と真偽をとる属性を有するデータ間の相関を見い出すための手法を提案するものである。

【0008】特に、（1）真偽をとる属性が真であるデータの割合がある定められた値以上であって、含まれるデータ数が最大となるようなルールであるサポート最大化ルールや、（2）最低限含まれるデータ数が定められた場合、真偽をとる属性が真であるデータの割合が最大となるようなルールであるコンフィデンス最大化ルール、

（3）取り出される領域内部と外部との分割を考えた時に、分割前の情報量と比較した分割後の情報量の増分を最大化するルールである最適化エントロピールール、

（4）領域内外の分割を考えた時に、内外の「標準化された真偽の割合の平均からのずれ」の二乗和を最大化するルールである最適化インタクラスバリエーションルールを満たすような範囲（領域）を導出可能とすることも目的である。

【0009】さらに、上記のようなデータ間の相関を実時間内に行うことができるような手法を提供することも目的である。

【0010】また、データ間の相関を人間に見やすい形

で提示することも目的である。そして、多くの相関の状態を可視化することにより、使用する人間の選択の幅を増大させることも目的とする。

【0011】

【課題を解決するための手段】通常、解析対象物は多くの数値属性を有する。この中から2つの数値属性を選び、また、1つの真偽をとる属性について、以下のステップを行うことにより、上記の目的を達成するものである。すなわち、

(1) 2つの数値属性により平面を構成し、この平面をメッシュ分割し、各メッシュ（バケットともいう）内のデータ数及び真偽をとる属性が真となったデータの数をカウントする。このような平面は、各メッシュをピクセルとした場合、データ数が濃淡度、真偽をとる属性が真となるデータの数が彩度に該当するような、平面画像として捉えることができる。

(2) 所定の条件 θ に従い、平面の1つの軸に凸な領域である許容イメージを切り出し、データの相関の強い部分を見出す。先に述べたように、平面を画像として捉え、平面の1つの軸に凸であるという条件を満たす部分画像（部分領域）を許容イメージとして切り出す。

(3) 切り出した許容イメージとなる領域が、先に述べたようなサポート最大化ルール等の条件を満たしていれば、その領域をユーザに提示する。また、データベースからその領域に含まれるデータの必要な属性を引き出すことも、必要に応じて行う。

【0012】なお、切り出された領域を、そのままユーザに提示したり、複数の領域を切り出した場合には、それを動画として可視化することにより、所望の結合ルールを見出し易くすることもできる。

【0013】また、一旦領域を切り出した後に、それ以外の結合ルールを見出すべく、切り出された領域について、彩度を平均化し、再度切り出しステップを実行することも可能である。

【0014】最初に述べたような例の場合、流動性預金残高の軸と、年齢の軸を設け、その平面を適当なメッシュに分割する。そして、各メッシュについて該当する顧客の数と、定期預金残高200万円以上の顧客の数をカウントする。そして、例えば顧客全体の20%が入り且つ定期預金残高200万円以上である顧客の割合が最大となるような許容イメージである領域の切り出しを行うことにより、コンフィデンス最大化ルールを得ることができる。

【0015】また、例えば定期預金残高200万円以上の顧客割合が10%で最大の顧客数を有する領域を切り出すことにより、サポート最大化ルールを得ることができる。

【0016】以上述べた事項をまとめると、2種類の数値属性と、1種類の真偽をとる属性を含むデータを有するデータベースにおいて、データ間の結合ルールを導き

出す方法であって、2種類の数値属性に対応する2つの軸を有し且つ $N \times M$ 個のバケットに分割されている平面の各バケットに対応して、当該バケット（座標 (i, j) ）に属するデータの数 $u(i, j)$ 及び真偽をとる属性が真であるデータの数 $v(i, j)$ を記憶する平面構成ステップと、条件 θ を入力するステップと、

【数17】

$$\sum_{(i,j) \in S} g(i,j) = \sum_{(i,j) \in S} (v(i,j) - \theta u(i,j))$$

10 を最大にするようなバケットの領域 S を前記平面から切り出す領域切出ステップと、切り出された領域 S 内に含まれるデータを出力するステップとを含む。

【0017】また、入力された前記条件 θ とは異なる第2の条件 θ_2 を入力するステップと、

【数18】

$$\sum_{(i,j) \in S_2} g(i,j) = \sum_{(i,j) \in S_2} (v(i,j) - \theta_2 u(i,j))$$

を最大にするようなバケットの第2の領域 S_2 を平面から切り出すステップと、

20 【数19】

$$\theta_3 = \frac{V(S_2) - V(S)}{U(S_2) - U(S)}$$

（領域 S_2 に含まれ且つ真偽をとる属性が真であるデータの数を $V(S_2)$ 、領域 S に含まれ且つ真偽をとる属性が真であるデータの数を $V(S)$ 、領域 S_2 に含まれるデータ数を $U(S_2)$ 、領域 S に含まれるデータ数を $U(S)$ とする。）を第3の条件として、

【数20】

$$\sum_{(i,j) \in S_3} g(i,j) = \sum_{(i,j) \in S_3} (v(i,j) - \theta_3 u(i,j))$$

30

を最大にするようなバケットの第3の領域 S_3 を平面から切り出すステップとを含むようにすることも考えられる。このような処理は、最初の条件 θ で、初期の目的のルールを導き出せなかった場合に有用である。

【0018】さらに、切り出された領域 S 内の各バケットの $v(i, j)$ ・ $u(i, j)$ が、平面全体のデータ数に対する平面全体の真偽をとる属性が真であるデータ数の割合に等しくなるよう $v(i, j)$ を変更するステップと、当該変更された $v(i, j)$ を用いて、入力された条件 θ_4 に従い、

【数21】

$$\sum_{(i,j) \in S_4} g(i,j) = \sum_{(i,j) \in S_4} (v(i,j) - \theta_4 u(i,j))$$

を最大にするようなバケットの第4の領域 S_4 を切り出すステップとをさらに含むようにすることも考えられる。このようにすると、二次的な相関ルールを導き出すことができる。

【0019】また、先の平面構成ステップが、複数のデータから、 X 個のデータをランダムサンプリングするステップと、サンプリングされたデータを各数値属性につ

いてソートし、 $X \cdot i \cdot N$ ($i = 1, 2, \dots, N$) 番目に該当する数値及び $X \cdot n \cdot M$ ($n = 1, 2, \dots, M$) 番目に該当する数値を記憶するステップと、記憶された数値を基準にして、複数のデータを $N \times M$ 個のバケットに入れるステップとを含むようにすることも考えられる。このようにすると、各バケットにデータを高速に割り振ることができる。

【0020】ここで、

【数22】

$$g(i, j) = v(i, j) - \theta u(i, j)$$

を評価値とし、先の領域切出ステップが、平面の各列において、少なくとも1のバケットを含み且つ評価値が最大となるバケットの範囲を求める列内範囲導出ステップと、平面内の任意のバケット(座標(m, t))の前列にあるもう1つのバケット(座標(m-1, 1))を含み且つ当該前列(第1列)までで最大の評価値を有する領域と、列内範囲導出ステップにより導出される、前記任意のバケット(座標(m, t))及び前記もう1つのバケットと同一行であって前記任意のバケットと同一列にあるバケット(座標(m, 1))を含む評価値が最大となる範囲とを加えた領域を加算領域とする時、各前記任意のバケットに対し、それ自身及び当該任意のバケットを含む加算領域全体として評価値を最大化する前記もう1つのバケットを検出し、当該加算領域の評価値と共に記憶する検出記憶ステップと、平面全体で加算領域の評価値が最大となる前記任意のバケットを検出し、検出記憶ステップで検出*

$f(U(S), V(S))$

$$\begin{aligned} &= -V(S) \log \frac{V(S)}{U(S)} - (U(S) - V(S)) \log \frac{U(S) - V(S)}{U(S)} \\ &\quad - (V_{sum} - V(S)) \log \frac{V_{sum} - V(S)}{U_{sum} - U(S)} \\ &\quad - (U_{sum} - V_{sum} - U(S) + V(S)) \log \frac{U_{sum} - V_{sum} - U(S) + V(S)}{U_{sum} - U(S)} \end{aligned}$$

(U_{sum} は平面全体のデータ数、 V_{sum} は平面全体に含まれる真偽をとる属性が真のデータの数)を計算し、その値を領域Sに対応して記憶するエントロピ計算ステップと、条件を変更して先の領域切出ステップとエントロピ計算ステップを実行するステップと、 $f(U(S), V(S))$ ※

$$\begin{aligned} f(U(S), V(S)) = & U(S) \left(\frac{V(S)}{U(S)} - \frac{V_{sum}}{U_{sum}} \right)^2 \\ & + (U_{sum} - U(S)) \left(\frac{V_{sum} - V(S)}{U_{sum} - U(S)} - \frac{V_{sum}}{U_{sum}} \right)^2 \end{aligned}$$

を計算し、その値を領域に対応して記憶するインタクラスバリエーション計算ステップと、条件を変更して先の領域切出ステップとインタクラスバリエーション計算ステップとを実行するステップと、 $f(U(S), V(S))$ を最大化する領域Sを出力するステップとを含むようにする

* された前記もう1つのバケットを用いて、領域Sを導き出すステップとを含むようにすることができる。

【0021】また、切り出されるべき領域に含まれる最低限のデータ数である最小サポート数 U_{min} を入力するステップと、切り出された領域Sに含まれるデータ数 $U(S)$ と最小サポート数 U_{min} と比較するステップと、この比較の結果、 $U_{min} \leq U(S)$ であれば、当該領域Sを切り出されるべき領域として出力するステップと、前記比較の結果、 $U_{min} > U(S)$ 又は $U_{min} < U(S)$ の場合には、新たな条件 θ_0 にて、先の領域切出ステップを実施するステップとを含むようにすることが考えられる。これにより、コンフィデンス最大化ルールを導出することができる。

【0022】さらに、切り出されるべき領域における真偽をとる属性が真であるデータの数の割合 $minconf$ を入力するステップと、切り出された領域Sが、 $minconf \leq V(S) / U(S)$ ($U(S)$ は領域Sに含まれるデータ数、 $V(S)$ は領域Sに含まれ且つ真偽をとる属性が真であるデータの数)である場合には、当該領域Sを出力するステップと、 $minconf < V(S) / U(S)$ 又は $minconf > V(S) / U(S)$ である場合には、新たな条件 θ_0 にて領域切出ステップを実施するステップとを含むようにすることも考えられる。これによりサポート最大化ルールを導出することができる。

【0023】ここで、切り出された前記領域Sに対し、【数23】

※ (S) を最大化する領域Sを出力するステップとを含むようにすれば、最適化エントロピ領域を見つけることができる。

【0024】また、切り出された前記領域Sに対し、【数24】

$$\begin{aligned} &= -V(S) \log \frac{V(S)}{U(S)} - (U(S) - V(S)) \log \frac{U(S) - V(S)}{U(S)} \\ &\quad - (V_{sum} - V(S)) \log \frac{V_{sum} - V(S)}{U_{sum} - U(S)} \\ &\quad - (U_{sum} - V_{sum} - U(S) + V(S)) \log \frac{U_{sum} - V_{sum} - U(S) + V(S)}{U_{sum} - U(S)} \end{aligned}$$

ことも考えられる。これにより、最適化インタクラスバリエーション領域を切り出すことができる。

【0025】以下の説明を理解すれば、上述の方法を実施するような装置を作成すること、またこのような方法をコンピュータに実施させるプログラムを格納した記憶

装置又は記憶媒体を作成することは容易に実施できるであろう。

【0026】

【発明の実施の形態】まず、本発明の各ステップがどのように実施されるかを示す。

(1) 平面構成ステップ

図1に、平面構成ステップのフローを示す。ステップ100にて処理が開始し、まずデータ集合Pからデータのランダムサンプリングを行う(ステップ110)。そして、サンプリングされたデータ $p_i(x_i, y_i)$ (x_i, y_i はデータの2つの数値属性の値を示す。)の x_i, y_i ごとにソートを行う(ステップ120)。この時、2つの数値属性の各々に対応するように2軸を有する平面を考え、その平面を各軸ごとにN個のバケットに分割する。すなわち、平面上には N^2 個のバケットが存在することとなる。そして、 x_i, y_i ごとに、 $i \cdot X \div N$ ($i = 1, 2, \dots, N-1$)番目の値を見つけ出す(ステップ130)。Xはサンプリングされたデータの数である。このようにすると、平面上の各列及び各行に属するデータの数は一致する。そして、 x_i, y_i ごとの $i \cdot X \div N$ ($i = 1, 2, \dots, N-1$)番目の値を用いると、各バケットの境界数値が分かるので、それを用いて各バケット(座標 (i, j))に入るデータ p_i の数 $u(i, j)$ と、真偽をとる属性が真であるデータの数 $v(i, j)$ を、各バケットごとにカウントする(ステップ140)。各バケットごとに、 $u(i, j)$ と $v(i, j)$ を記憶する(ステップ150)。言い換えると、 $u(i, j)$ 行列と $v(i, j)$ 行列が生成されたということができる。

【0027】上述のようにランダムサンプリングを行うのは、全てのデータをソートしていると時間がかかるからである。但し、ソートしてもよい場合もある。また、ランダムサンプリングで取り出されるデータの数は、 $30N$ から $50N$ ぐらいが好ましい。また、2軸ともN個に分割する例を示したが、異なる数に分割することも可能である。典型的な例で、Nは100ぐらいである。

【0028】以上述べたのは一例であって、他の方法を用いてもよい。例えば、各バケットの境界数値については予め定めた値を用いても良い。また、データ値に対して均等に分割することも、また対数的に分割することも可能である。

【0029】また、後の処理のため以下のような処理(図2)を行っておくと、さらに全体の処理が高速化される。すなわち、 $u(i, j)$ と $v(i, j)$ の行数(N_y)と列数(N_x)を調べる(ステップ210)。そして、先に求めた $u(i, j)$ と $v(i, j)$ を用いて、新たに以下のような $u'(i, j)$ と $v'(i, j)$ という行列を作成する(ステップ220)。

【数25】

$$u'(i, j) = \sum_{k=0}^i u(k, j)$$

(9)

特開平9-179883

16

【数26】

$$v'(i, j) = \sum_{k=0}^i v(k, j)$$

【0030】この $u'(i, j)$ と $v'(i, j)$ は、後々数多く計算することとなる目的関数、

【数27】

$$g(i, j) = v(i, j) - \theta u(i, j)$$

の和計算を以下のように簡単化するために用意する。

【数28】

$$\begin{aligned} \sum_{k=i}^j g(k, m) &= \sum_{k=i}^j v(k, m) - \theta \sum_{k=i}^j u(k, m) \\ &= (v'(j, m) - v'(i-1, m)) \\ &\quad - \theta (u'(j, m) - u'(i-1, m)) \end{aligned}$$

【0031】さらに、

【数29】

$$U_{\text{sum}} = \sum_{\text{全体}} u$$

【数30】

$$V_{\text{sum}} = \sum_{\text{全体}} v$$

も後によく用いるので用意する。

【0032】以上のような準備をすれば、以下の領域切り出しステップが高速になる。なお、上述のように作成された平面は、平面“画像”と考えることもできるので、イメージという文言を用いる場合がある。同様に、バケットはピクセルということもある。

【0033】(2) 領域切り出しステップ

このステップにおいては許容イメージである領域を先に作成した平面(平面画像)から切り出すものである。この許容イメージは、先に述べたように1つの軸方向に凸であるイメージを言う。より正確に言うと、X軸方向へ単調な2本の曲線で囲まれる連結イメージを言う。このことは、図3に例を示す。左には、幅1のY軸方向に伸びる帯で切ると必ず連結している例を示しており、このようなイメージを許容イメージという。また、右には、先の帯で切ると連結していない例が示されており、このようなイメージを含めて切り出そうとすると、その問題はNP困難になってしまう。

【0034】このような許容イメージに問題を限定すると、Y軸方向に伸びる帯は必ず連結しているので、ダイナミックプログラミングを用いて、それらを順々に連結していけばよい。但し、もう1つのパラメータを指定する必要がある。例えば、濃淡のみの画像の場合にはピクセル数(バケット数)でもよい。例えばユーザにピクセル数を指定させ、そのピクセル数で且つ濃淡度の総和が最大になる許容イメージを取り出すことができる。

【0035】全体の許容イメージの連結性は、以下の事項を考えると分かる。すなわち、k個のピクセル(バケ

50

17

ット)を含み、m列より左のピクセルからなり、(m, t)の位置を含むイメージ(図4参照のこと)を考え、この時に濃淡値の総和をf(k, m, t)とすると、こ*

$$f(k, m, t) = \max_{t, l \in I} \{f(k-|I|, m-1, l) + \sum_{i \in I} q(i, m)\}$$

【0036】ここで、IはX軸上m列目の連続区間(範囲)であり、t, lを含む。q(i, j)は、座標(i, j)の濃淡度を示す。また、f(k-|I|, m-1, l)を計算する時に用いたX軸上m-1列目の連続区間(範囲)はlを含むので、lと連結している。従って、帰納法よりf(k, m, t)を求めるまでに用いた連続区間全体は連結していることが保証されている。

【0037】このようにすると、O(N⁶)のオーダーで計算が可能となり、N²=n(nはピクセル数)とするとO(n³)となり、実時間で計算可能となる。

【0038】以上濃淡のみの画像においてピクセル数(バケット数)を特定する場合の例を説明したが、本発明では濃淡のみではないので、他の方法を用いる。すなわち、図5に示すような、横軸が切り出される領域Sに含まれるデータ数U(S)、縦軸が切り出される領域Sに含まれ且つ真偽をとる属性が真であるデータの数V(S)であるような平面を考える。データ数と真偽をとる属性が真であるデータの数の組み合わせは多数存在するので、この平面には多数の点が存在することになるが、この点のうち、凸包を構成する点を特に用いる。すなわち、この凸包を構成する点をつなぐことにより曲線を構成し、この曲線に対し傾きθを有する直線を上から下ろして行き、最初にこの曲線と接する点を求め、この時のイメージを出力するという方法を用いる。この時の許容イメージをフォーカス・イメージ(focused image)という。フォーカス・イメージを図5では黒丸で示している。また、直線を下ろしていくような方法をハンドプロブという。このように、本発明では傾きθを入力するような方法を用いる。

【0039】このように凸包上の点のみ取り扱うのは、※

$$\begin{aligned} f(m, t) &= \max_{t, l \in I} \{f(m-1, l) + \sum_{i \in I} g(i, m)\} \\ &= \max_l \{f(m-1, l) + \underbrace{\max_{t, l \in I} \sum_{i \in I} g(i, m)}_A\} \end{aligned}$$

この数33のAは、t, lを含む連続区間(範囲)全体で数33のBを最大化する連続区間(範囲)lを見つけることを意味する。

【0042】このlをcover(t, l)と記述することとする。いま、t ≤ lを仮定すると、次の定義されるlow(t), high(l)を用いれば、

【数34】

18

※これは以下の数式を満たすものである。

【数31】

※コンフィデンス最大化ルール、サポート最大化ルールは、凸包上に必ず存在するわけではないが、近似解としては十分な点を出力することができ、また最適化エントロピー・ルール及び最適化インタクラスバリエーション・ルールについては、この凸包上に必ず存在するからである。もし、コンフィデンス最大化ルール及びサポート最大化ルールの厳密解を解くとすると、実時間には計算が終了しないので、近似解であっても十分に有効な結果を計算できる。

【0040】上記のように傾きθの直線を下ろしていくということは、直線y = θx + QのY切片であるQを減少させることであり、言い換えれば、Q = V(S) - θU(S)を最大にするU(S)をX座標に有する点を求める問題となる。よって、

$$\begin{aligned} \text{【数32】} \\ \max Q &= \max \left\{ \sum_{(i,j) \in S} v(i,j) - \theta \sum_{(i,j) \in S} u(i,j) \right\} \\ &= \max \sum_{(i,j) \in S} g(i,j) \end{aligned}$$

と変形される。

【0041】では、この数32をどのように解くかを考える。基本的には先に述べたダイナミック・プログラミングを用いた手法を用いる。まず、m列目より左のバケットからなり、(m, t)の位置のバケットを含む領域を考え、この中で目的関数である数27を最大化したものをf(m, t)とすると、以下の条件を満たすすなわち、

【数33】

$$\begin{aligned} \text{cover}(t, l) &= [\text{low}(t), t] \cup [t, l] \cup [l, \text{high}(l)] \end{aligned}$$

となる。但し、low(t)は連続区間[i, t]全体で数33のBが最大となるiであり、high(l)は連続区間[l, j]全体で数33のBが最大となるjを言う。

【0043】このlow(t)やhigh(l)はダイナミック・プログラミング中では何度も用いられるの

で、low(t)やhigh(l)を高速に求めることができれば有効である。このため、連続区間[i, j]中の数33のBが要素K(i, j)に入る行列Kを作る。但し、i>jの場合にはK(i, j)=(i-j)x(xは十分、その絶対値が大きい負の数。例えば、u(i, j)を全体について加算した値より絶対値が大きいならば十分である。)としておく。すると行番号1において最大値を有する列の列番号がhigh(l)となる。よって、すべてのl∈[1, N]についてhigh(l)を求める問題は、Kの各行の最大値の列番号を求める問題となる。このような計算は、O(N)の計算量で行える。

【0044】最大値の列番号を求める場合に最大値かどうかを判断するための行列の要素の比較を行う。この比較は、先に数25及び数26にて求めたu'及びv'を用いると簡単に済む。

【0045】このKで各行の最大値に注目すると、行番号が大きくなるにつれて最大値の列番号は単調に増加する。但し、同じ行に最大値が複数個ある時には、左端のみを考える。このような行列を「単調な行列」と呼ぶ。*20

$$f(m, t) = \max_l \{ f(m-1, l) + \sum_{i \in \text{cover}(t, l)} g(i, m) \}$$

【0048】f(m, t)を最大にするような、イメージを見い出すには、f(m, t)の計算をX軸に垂直な帯について順に行い、その帯を記憶しておき、それらを連結すれば求まる。

【0049】さらに高速化するには、

【数36】

$$M(t, l) = f(m-1, l) + \sum_{i \in \text{cover}(t, l)} g(i, m)$$

を要素として有する行列Mを作り、行番号tの最大値がf(m, t)となる。Mは先に説明した完全単調な行列であり、すべてのtについてf(m, t)はO(N)で計算することができる。よって、すべてのmについてf(m, t)を計算するには、O(N²)の計算量が必要となる。

【0050】以上詳細を述べたが、必要なステップを以下に示しておく。

(1) 全てのX軸に垂直な帯についてlow(t), high(l)を計算しておく。

(2) low(t), high(l)によりcover(t, l)が求まるので、数36を要素とする行列Mを計算する。

(3) 行列Mの各行の最大値を求め、その値をf(m, t)として記憶する。

(4) イメージ全体を把握するために、行列Mの各行の列番号lをs(m, t)に入力する。

(5) f(m, t)を最大にするm, tを求め、(4)で作ったs(m, t)及びs(m, t)に記憶されているlを用いて前列のlow(t), high(l)でも

* 証明は省略するがこの行列Kは「完全単調な行列」(任意の部分行列が単調行列であるような行列。)でもある。Kの一例を図6に示し、斜線部が各行の最大値である。単調な行列の全ての行の最大値を有する列番号を計算するにはO(N log N)の計算量が必要である。この完全単調な行列の各行の最大値の列番号を求めるアルゴリズムは周知であり、例えば、「計算幾何学」浅野哲夫著、朝倉書店、1990年9月の第4章「計算幾何学の基本的技法」に記載されている。

10 【0046】同様にしてlow(t)を計算する場合には、連続区間[i, j]中の数33のBが要素L(j, i)に入る行列Lを作り、行番号tにおける最大値を有する列の列番号を求めれば、low(t)になる。今度は上三角部分(i>j)を-∞とする。このような行列も完全単調な行列である。

【0047】このようにして求めたlow(t)及びhigh(l)を用いれば、cover(t, l)が求まり、数33の変形である以下の式が計算可能となる。

【数35】

って、イメージを把握する。

【0051】このステップを図7及び図8に示す。ステップ610で開始した処理は、ステップ620においてm=1の初期化を行う。そしてm=Nx-1であるかを判断する(ステップ630)。これはループをNx回繰り返すためであり、Nxはu(i, j)とv(i, j)の列数である。

30 【0052】この後に、

【数37】

$$K(i, j) = \begin{cases} (i, j)x & (i > j) \\ \sum_{h \in [i, j]} g(h, m) & (i \leq j) \end{cases}$$

を計算しておき、この行列の各行の最大値を求め、その列番号jをhigh(m, i)とする(ステップ640)。これにより、前記平面(平面画像)のm列目のhigh(i)が求まった。

40 【0053】また、

【数38】

$$L(i, j) = \begin{cases} -\infty & (j > i) \\ \sum_{h \in [j, i]} g(h, m) & (j \leq i) \end{cases}$$

を計算しておき、この行列の各行の最大値を求め、その列番号jをlow(m, i)=jとする(ステップ650)。これにより、前記平面(平面画像)のm列目のlow(i)が求まった。

50 【0054】その後にmを1インクリメントして(ステッ

ブ660)、ステップ630に戻る。このように、まず low と high を最初にすべて計算しておく。図8の計算を実行するごとに必要な low と high を計算するようにしてもよいが、この例のように一度に計算してしまってもよい。上述のように一度に計算した後に処理はXを介して図8に進む。

【0055】図8においてXから、 $f(m, t)$ を計算する。まず、図4のような前記平面の一番左の列について処理する。すなわち、

【数39】

$$f(1, t) = \sum_{i \in [\text{low}(1, t), \text{high}(1, t)]} g(i, 1)$$

を $t = 1$ から N_y について計算する(ステップ710)。 N_y は $u(i, j)$ と $v(i, j)$ の行数である。このようにすると、数36の第1項の初期値となるものが計算されたこととなる。

【0056】また、 $s(1, t) = -1$ としておく、これは、最初の列であるから、これ以上前の列には連結があり得ないことを明示するためである。

【0057】そして、 $m = 2$ 以降の値を計算するために、 $m = 2$ とし(ステップ720)、以下のループを $N_x - 1$ 回まわすため、 $m = N_x + 1$ かどうか判断する(ステップ730)。もし、 $m = N_x + 1$ でなければ、 $f(m-1, i)$ ($1 \leq i \leq N_y$) に負の値が一つでもある場合には、 $t = 1$ から N_y まで、

【数40】

$$f'(m, t) = \sum_{i \in \text{cover}_m(t, t)} g(i, m)$$

を計算する(ステップ742)。ここで、 $\text{cover}_m(t, t)$ は、 m 列目の $\text{cover}(t, t)$ の意味である。そして、

【数41】

$$M(t, l) = f(m-1, l) + \sum_{i \in \text{cover}_m(t, l)} g(i, m)$$

この行列Mの各行の最大値の列番号を求める(ステップ744)。但し、その列 i の最大値と $f'(m, i)$ と比較して大きい方を $f(m, i)$ とする。そして、 $s(m, t)$ には、 $f'(m, t)$ がMの t 行の最大値よりも小さいときには1を、そうでなければ-1を入れる。これは、前列までの連結を保存した方が切り出される領域の目的関数の値が大きくなるか、それとも前列までの連結を放棄した方が目的関数の値が大きくなるかという判断をしているのである。

【0058】このように $s(m, t)$ は連結状態を保存するためにあるので、ある (m, t) が決まれば、 $s(m, t)$ を手繰って遡り、領域がどのように連結するものかを後にみることができる。

【0059】この後に、 m を1インクリメントし(ステップ750)、ステップ730に戻る。繰り返しが全て終われば、 $f(m, t)$ を最大とする m, t が求まる。 $f(m, t)$ を作りながら、常に最大となる m, t を保持しておき、新たに作成された部分につき保持している m, t より大きな点を見出した場合には更新するようにしていけばよい。この m, t を用いて、 $s(m, t)$ から、前列の l が求まる。この l と t のうち小さい方を low に大きい方を high に入力する。例えば、 t の方が小さければ、 $\text{low}(m, t), \text{high}(m, t)$ が求まる。また、 $s(m-1, l)$ から、さらに前列の l' が求まるので、 $\text{low}(m-1, l), \text{high}(m-1, l')$ を求める。(ここでは、 l の方が l' より小さい。)これを繰り返していくと、イメージSの全体が分かる(ステップ760)。ある $s(m, t)$ にて-1が得られれば、その領域は終了する。これにて、傾き θ を入力した場合に、その θ に対応する許容イメージSを得ることができたので、処理を終了する(ステップ770)。

20 【0060】以上説明したように、傾き θ に対応する許容イメージを得ることができた。この条件 θ は、ユーザが入力するようにしてもよいし、また例えばサポート最大化ルールを求めるためにシステムにより設定された条件であってもよい。

【0061】(3) 出力ステップ

以上のように求まった許容イメージたる領域Sは、前記平面のどの部分を占めているかは、先のステップによりわかっているので、そのイメージSに属するデータを取り出すことになる。通常各データは、真偽をとる属性及び数値属性のみならず、他の属性も有しているから、例えばダイレクトメールを送るのであれば、住所氏名といった属性を取り出すようになる。ここまでくると、取り出すべきデータは特定されているから、通常のデータベースの検索に過ぎないので、これ以上詳しく述べない。当然、一旦フォーカス・イメージをその外形がよくわかるようにして、ユーザに提示するようにしてもよい。

【0062】以上のような各ステップを実施すれば、ある条件 θ に対する、データ間結合ルールの1つを求めることができる。しかし、この条件 θ をどのように設定するかということは、1つの問題である。通常、ある条件 θ 1つでは、問題の解決にならない場合が多い。以上の各ステップ、特に(1)平面構成ステップと(2)領域切り出しステップとをエンジンとして用い、どのように先に述べた4つの一般的なルール及び他のルール等を導き出すかを以下に示す。

【0063】A. ある区間に存在するフォーカス・イメージを求める場合

まず、幾つかの θ に対応する許容イメージ(ここではフォーカス・イメージ)たる領域Sを連続的に示し、動画を作成することにより、切り出されるイメージの大きさ

及び形状をユーザの判断により決定させる場合を考える。

【0064】この処理を図9に示す。ステップ800にて開始された処理は、まず $\theta 1$ を入力することにより、上述したプロセスにてフォーカス・イメージS1を見つけ出す(ステップ810)。また、ユーザに $\theta 2$ を入力させ、同様にフォーカス・イメージS2を見つけ出す(ステップ820)。このようにして2つのフォーカス・イメージが求まると、それぞれに含まれるデータ数U(S1)、U(S2)及び真偽をとる属性が真であるデータの数V(S1)、V(S2)とを用いて、その中間にある、新たな傾き $\theta 3$ を計算する(ステップ830)。

【0065】このように新たな $\theta 3$ が求まれば、さらにこの $\theta 3$ に対応するフォーカス・イメージS3を求めることができる(ステップ840)。ここで、計算されたS3が既に求まっていれば、区間($\theta 1$, $\theta 2$)にはこれ以上のフォーカス・イメージは存在しない。よって、処理が終了する(ステップ880)。しかし、発見済みでなければ、 $\theta 2$ の代わりに $\theta 3$ を用いて、ステップ830以降を実行する(ステップ860)。すなわち、区間($\theta 1$, $\theta 3$)の間にあるフォーカス・イメージを見つけ出す。この場合、次々に中間の値を計算していくようにすることも可能である。また、ある程度の個数フォーカス・イメージが求まったところで計算を取り止めることもできる。さらに、もう1つ残った区間($\theta 3$, $\theta 2$)についてフォーカス・イメージを計算するために、 $\theta 3$, $\theta 2$ についてステップ830以降を実行する(ステップ870)。この場合も、この区間内に存在しているフォーカス・イメージを全て見つけ出すようにしてもよいし、所定の個数見つけ出したところで処理を終了してもよい。

【0066】このようにして、1つ又は複数のフォーカス・イメージを見つけ出すことができた。このように求まった複数のフォーカス・イメージを図10のように(a)、(b)、(c)と連続してユーザに提示するようなことも可能である。この図10の各々の斜線部分が切り出された領域(フォーカス・イメージ)である。

【0067】B. コンフィデンス最大化ルールの場合(図11及び図12)

この場合には、ルールの定義より最小限度のサポートminsup(全体のデータ数に対する包含されるデータ数の割合)を入力することとなる(ステップ910)。ここで、 $Umin = Usum * minsup$ を計算しておく。ここで図5をみると、最小限度サポートと記された縦の点線がこの値に対応する。そして、 $\theta = Vsum / Usum$ としてフォーカス・イメージS1を求める(ステップ920)。このS1に含まれるデータ数U(S1)の値により3つの場合に分けられる(ステップ930)。すなわち、

(1) $U(S1) \leq Umin$ であれば、当然S1が解として出力され、処理は終了する(ステップ990)。(2)

$U(S1) < Umin$ であれば、 $\theta = 0$ としてフォーカス・イメージS2を計算する(ステップ940)。この結果、 $U(S2) \leq Umin$ であれば(ステップ950)、当然S2が解として出力され、処理は終了する。ここで、 $S1 = S2$ はminsupが1より小さければあり得ない。但し、 $S1 = S2$ であれば解はないので、解なしを出力する。そうでなければ、XXにて図12に移動する。

(3) $U(S1) > Umin$ であれば、 $\theta = 1$ としてフォーカス・イメージS2を計算する(ステップ960)。S1 = S2であるならば、S1より高いコンフィデンスを有するフォーカス・ポイントは存在しないので、S1(当然S2も同様)を最良解として出力し、処理を終了する(ステップ970)。同様に、 $U(S2) \leq Umin$ であれば、S2を出力して処理を終了する。ステップ970にて、先に示した条件を満たさないとされた場合には、XXを介して図12に移動する。

【0068】図12では、XXから始まり、新たな条件 θ を求め、この θ に対するフォーカス・イメージSを計算する(ステップ1400)。この θ は

$$\theta = (V(S2) - V(S1)) / (U(S2) - U(S1))$$

にて計算される。そして、 $S1 = S$ 又は $S2 = S$ であるならば、(S1, S2)の間にはこれ以上フォーカス・イメージは存在しないので、コンフィデンスの高いS2が最良解として出力され、処理を終了する(ステップ1410)。また、 $U(S) \leq Umin$ であるならば、Sを出力し、処理を終了する。

【0069】ところが、 $U(S) < Umin$ であると(ステップ1420)、まだ処理が必要なので、 $S1 = S$ として(ステップ1430)、ステップ1400に戻る。同様に、 $U(S) > Umin$ であるならば、 $S2 = S$ として(ステップ1440)、ステップ1400に戻る。

【0070】これを繰り返すことにより解が見つけられる。図5を参照すると、先に説明した最小限度のサポートの右側、濃く塗られた部分に解の存在する範囲がある。そして、この図5の場合には、凸包の内部の白丸の点が厳密解となるが、本発明ではハンド・プロープにて得られた近似解が出力される。見つけられた解は、ユーザに提示されるようにしてもよいし、そのフォーカス・イメージに属するデータの必要な属性を出力するようにしてもよい。

【0071】C. サポート最大化ルールの場合(図13、図14)

このルールの場合、定義より、最小限度のコンフィデンス、すなわち含まれるデータ数に対する真偽をとる属性が真である割合を入力する(ステップ1110)。図5の場合、最小限度のコンフィデンスと示され、原点から引かれた点線がこれに該当する。次に、 $minconf \leq Vsum / Usum$ であるかを判断する(ステップ1120)。この条件に合致する場合には、全ての凸包上の点は条件に

合致するので、 $\theta = 0$ としてフォーカス・イメージを求め、それを出力する。(ステップ1130)。これに対し、ステップ1120で $\text{minconf} > V_{\text{sum}} \cdot U_{\text{sum}}$ である場合には、 $\theta = V_{\text{sum}} \cdot U_{\text{sum}}$ としてフォーカス・イメージS1を求める(ステップ1140)。ここで、3つの場合が考えられる(ステップ1150)。すなわち、

(1) $\text{minconf} \equiv V(S) \cdot U(S1)$ である場合には、S1を出力して処理を終了する(ステップ1190)。
 (2) $\text{minconf} < V(S1) \cdot U(S1)$ である場合には、 $\theta = 0$ のフォーカス・イメージS2を求める(ステップ1160)。このS2は、 $V(S2) = V_{\text{sum}}$ であり、且つ $U(S2) = U_{\text{sum}}$ であるから、 $\text{minconf} \leq V(S2) \cdot U(S2)$ とはなり得ない。同様に、 $S1 = S2$ はあり得ない。よって、図14への移動するためにYに移る(ステップ1160)。(3) $\text{minconf} > V(S1) \cdot U(S1)$ である場合には、 $\theta = \text{minconf}$ のフォーカス・イメージS2を求める(ステップ1170)。このS2に対し、 $\text{minconf} = V(S2) \cdot U(S2)$ が成り立てば、S2は厳密解であるから、これを出力して処理を終了する。また、 $\text{minconf} < V(S2) \cdot U(S2)$ であれば、解は存在しないので、解なしを返して処理を終了する。一方、 $\text{minconf} > V(S2) \cdot U(S2)$ であるならば、図14の処理に移行するためにYに進む。

【0072】図14では、Yから処理が開始され、 $\theta = (V(S2) - V(S1)) \cdot (U(S2) - U(S1))$ としてフォーカス・イメージSを求める(ステップ1200)。この求められたフォーカス・イメージSに対し、(1) $\text{minconf} \equiv V(S) \cdot U(S)$ が成立する場合 *

$$f(x, y) = -y \log \frac{y}{x} - (x-y) \log \frac{x-y}{x} - (b-y) \log \frac{b-y}{a-x} \\ - (a-b-x+y) \log \frac{a-b-x+y}{a-x}$$

このxは $U(S)$ 、yは $V(S)$ 、aは U_{sum} 、bは V_{sum} である。このような条件においても、解は凸包上に存在することが分かったので、上述のステップを用いることができる。よって、 θ を変化させ、数42を最大化するフォーカス・イメージを求めればよい。

【0075】E. 最適化インタクラスバリエーション・ルールの場合

※40 【数43】

$$f(x, y) = x \left(\frac{y}{x} - \frac{b}{a} \right)^2 + (a+x) \left(\frac{b-a}{a-x} - \frac{b}{a} \right)^2$$

x、y、a、bは上述したものと同じである。このような条件においても、解は凸包上に存在することが分かったので、上述のステップを用いることができる。よって、 θ を変化させ、数43を最大化するフォーカス・イメージを求めればよい。

【0076】F. その他

以上述べたように、 $U(S)$ と $V(S)$ 上の凸包上の点

* 合には、このSを出力して処理を終了する(ステップ1210)。また、 $S1 = S$ 若しくは $S2 = S$ である場合には、これ以上S1とS2の間には解は無いので、S1を最良解として出力し、処理を終了する(ステップ1210)。これに対し、 $\text{minconf} < V(S) \cdot U(S)$ である場合には(ステップ1220)、 $S1 = S$ としてステップ1200に戻る(ステップ1230)。また、 $\text{minconf} > V(S) \cdot U(S)$ である場合には、 $S2 = S$ としてステップ1200に戻る(ステップ1240)。

10 【0073】以上のようにして、サポート最大化ルールが求められる。もう一度図5に戻ると、先に説明した最小限度のコンフィデンスとして示した点線より上の濃く塗られた範囲に解が存在する。そして、この例では凸包内の白丸の点が厳密解であるが、このように凸包内部の点は見つけ出すのに膨大な計算量を必要とするので、凸包上の点でサポートを最大にする近似解を出力するようにしている。先に述べたように、見出された近似解又は厳密解は、ユーザに提示してもよいし、フォーカス・イメージ内に含まれるデータの必要な属性値を出力するよう

20 【0074】D. 最適化エントロピー・ルールの場合
 最適化エントロピー・ルールとは、領域の内部と外部との分割を考えた時、分割前の情報量と比較した分割後の情報量の増分を最大化するルールである。よって、切り出された領域と平面全体のエントロピーのゲイン(以下の式)が最大となる領域を発見すればよい。

【数42】

※ 先に述べたように最適化インタクラスバリエーション・ルールとは、領域内外の分割を考えた時、内外の「標準化された真偽の割合の平均からのずれ」の二乗和を最大化するルールである。よって、切り出された領域と平面全体のインタクラスバリエーション(以下の式)が最大となる領域を発見すればよい。

※40 【数43】

$$f(x, y) = x \left(\frac{y}{x} - \frac{b}{a} \right)^2 + (a+x) \left(\frac{b-a}{a-x} - \frac{b}{a} \right)^2$$

に存在する又は存在すると近似できる場合には、上述したステップを用いれば高速にルールに該当する領域を導き出すことができる。

【0077】G. 二次的なルールの抽出

上述のプロセスを用いて1つのルールを見出した後に、二次的なルールを見つけて出すことができる。すなわち、切り出した1のフォーカス・イメージに属するv

(i, j)を除去し、 $v(i, j) \cdot u(i, j) = Vsum \cdot Usum$ となるように、 $v(i, j)$ を変更し、それから新たに領域切り出しステップを行うのである。

【0078】以上、本発明における処理のプロセスを説明した。このような処理プロセスは、コンピュータ・プログラムによって実現し、実行するようにしてもよい。例えば、図15のような通常のコンピュータ・システムにおいて実行できるようなプログラムにすることもできる。処理プログラムは、HDD1050に格納され、実行時にはメインメモリ1020にロードされ、CPU1010によって処理される。また、HDD1050はデータベースをも含んでおり、処理プログラムはそのデータベースに対するアクセスを行う。最初の平面やフォーカス・イメージ(図17)は、表示装置1060によってユーザに提示される。ユーザは、入力装置1070にてフォーカス・イメージの選択や、データ出力の命令を入力する。このような入力装置には、キーボードやマウス、ポインティング・デバイスやディジタイザを含む。さらに、出力結果を補助記憶装置であるFDD1030のフロッピー・ディスクに記憶したり、また新たなデータをFDD1030から入力することもできる。さらに、CD-ROMドライブ1040を用いて、データを入力することもできる。

【0079】さらに、本発明の処理プロセスを実現したコンピュータ・プログラムは、フロッピー・ディスクやCD-ROMといった記憶媒体に記憶して、持ち運ぶことができる。この場合、通常のデータベース検索プログラムのデータ取り出し部分や、表示装置1060に表示するだけの処理を行うプログラムは、すでにHDD1050に記憶されている場合もある。よって、それ以外の部分が、上記のような記憶媒体にて流通することは通常行われる事項である。

【0080】また、本発明の処理を専用に行うような装置を設けてもよい。例えば、図16のような装置が考えられる。平面構成部1310は、データベース1300及び切出部1320に接続されており、制御部1340からの命令を受付ける。また、切出部1320は、出力部1330及び表示装置1350に接続されており、制御部1340からの命令を受付ける。また、切出部1330はデータベースにも接続を有している。制御部1340は、入力部1360に接続され、入力部1360により指示された処理の種類により平面構成部1310及び切出部1320を制御する。

【0081】この装置の簡単な動作を説明する。平面構成部1310は、先に説明した平面構成ステップを実行する部分である。このように平面構成部1310は、データベースに記憶されたデータを用いて先に示した平面を構成し、切出部1320に出力する。切出部1320は、制御部1340からの命令に従って、切り出しのためのパラメータである θ をセットする。セットされた θ

に従って切出部1320は、先に述べた切出ステップを行い、フォーカス・イメージを切り出す。そして、表示装置1140に出力し、ユーザに命令されれば、切り出されたフォーカス・イメージ内に属するデータをデータベース1300から取り出し、出力部1330に引き渡す。出力部1330は、適当な形式でユーザ所望のデータを出力する。また、ユーザは、例えば入力部1360からコンフィデンス最大化ルールを解くように命じ、最小限度のサポートを入力する。すると、制御部1340は先に示した処理Bを行うように、条件 θ を設定し、切出部1320に出力する。そして、命じられたコンフィデンス最大化ルールに合致するような領域を解くべく、条件 θ を変化させる等の処理を行う。先に述べたサポート最大化ルール(処理C)や、最適化エントロピー・ルール(処理D)、最適化インタクラスバリエーション・ルール(処理E)、その他凸包上に位置する領域を切出す処理Fに適した条件 θ を切出部1320に渡す処理を制御部1340は行う。ユーザは入力部1360から処理の種類や、先に述べたような条件(θ のみならず、 \minconf , \minsup も)を入力する。また、制御部1340は、上述の処理Gを行うために平面構成部1310に、切り出したフォーカス・イメージの $V(i, j)$ を除去する等の処理を命じる。

【0082】以上、本発明を専用の装置にする一例を示したが、本発明はこれに限定されるものではない。例えば、切出部1320の出力は、出力制御部を介して出力部1160及び表示装置1140に出力されるようにしてもよいし、この場合出力制御部からデータベースを参照してデータを取り出すようにしてもよい。

【0083】以上は、通常データが有するK個の数値属性のうち2項を選択し、それらの数値属性間の壮観を見つける処理であったが、数27を目的関数とし、n次元空間の領域を切り出すことができれば、n次元の探索に拡張することができる。

【0084】

【効果】以上述べたように、2項の数値属性と真偽をとる属性を有するデータ間の相関を見い出すことができた。

【0085】(1)サポート最大化ルール、(2)コンフィデンス最大化ルール、(3)最適化エントロピー・ルール、(4)最適化インタクラスバリエーション・ルールを満たすような範囲(領域)を導出可能とすることもできた。

【0086】さらに、上記のようなデータ間の相関を実時間内に行うこともできた。

【0087】また、データ間の相関を人間に見やすい形で提示することもできた。

【0088】例えば、ある割合以上で、例えばアウトドアスポーツに興味を示す(真偽をとる属性に相当する)、できるだけまとまった領域に入る顧客を知ること

ができるので、その条件に合致する多くの顧客に知ってもらいたいダイレクトメールの宛て先を知るのに用いることができる。(サポート最大化ルール)

【0089】一定数以上の顧客を含む、例えば定期預金残高200万円以上の顧客割合が最も高いところを知ることができるので、顧客を絞りこみつつ、有効な宣伝活動を行うことができる。(コンフィデンス最大化ルール)

【図面の簡単な説明】

【図1】平面構成ステップのフローを示す図である。

【図2】領域切り出しステップのための前準備のフローを示す図である。

【図3】許容イメージを説明するための図である。

【図4】許容イメージの連結性を説明するための図である。

【図5】U(S)、V(S)平面の説明をするための図である。

【図6】完全単調な行列の説明をするための図である。

【図7】切出ステップの一部を示すフローチャートである。

【図8】切出ステップの一部を示すフローチャートである。

【図9】複数のフォーカス・イメージを見つけ出す処理のフローを示す図である。

*【図10】フォーカス・イメージの一例を示した図である。

【図11】コンフィデンス最大化ルールを導出するための処理の一部を示すための図である。

【図12】コンフィデンス最大化ルールの導出するための処理の一部を示すための図である。

【図13】サポート最大化ルールの導出するための処理の一部を示すための図である。

【図14】サポート最大化ルールの導出するための処理の一部を示すための図である。

【図15】通常のコンピュータ・システムで本発明を実施した場合の装置構成の一例を示す図である。

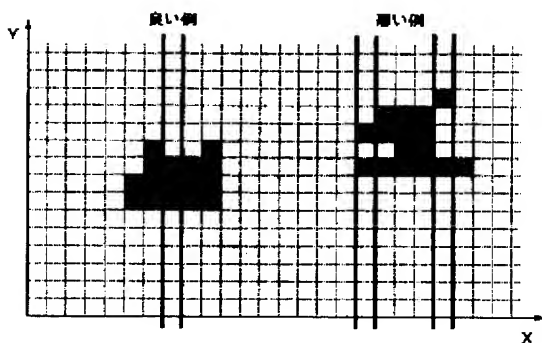
【図16】本発明を専用の装置で実施した場合のブロック図である。

【図17】図15の表示装置の表示例を示す図である。

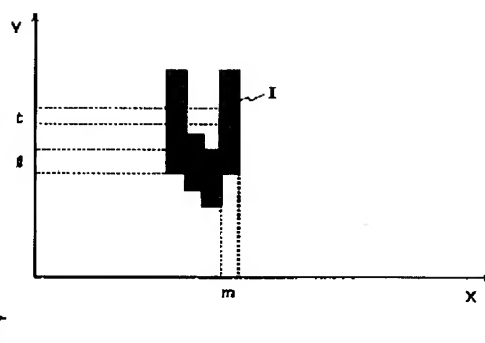
【符号の説明】

1010	CPU	1020	メインメモリ
1030	FDD	1040	CD-ROMドライブ
1050	HDD	1060	表示装置
1070	入力装置		
1310	濃淡画像構成部		
1300	データベース	1320	切出部
1350	表示装置	1130	入力部
1330	出力部	1340	制御部

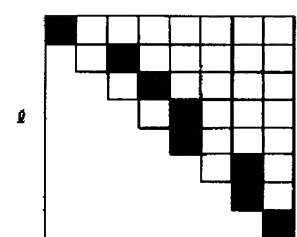
【図3】



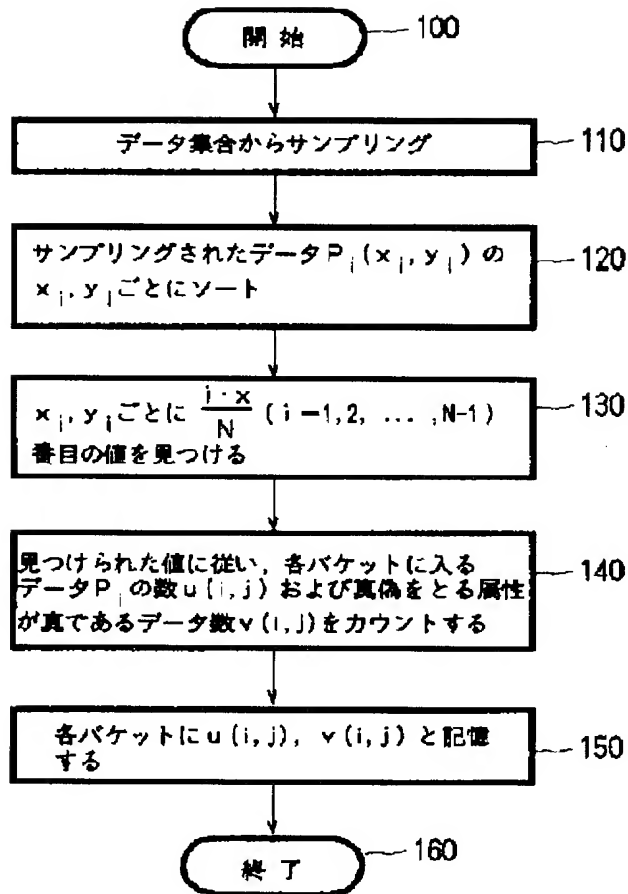
【図4】



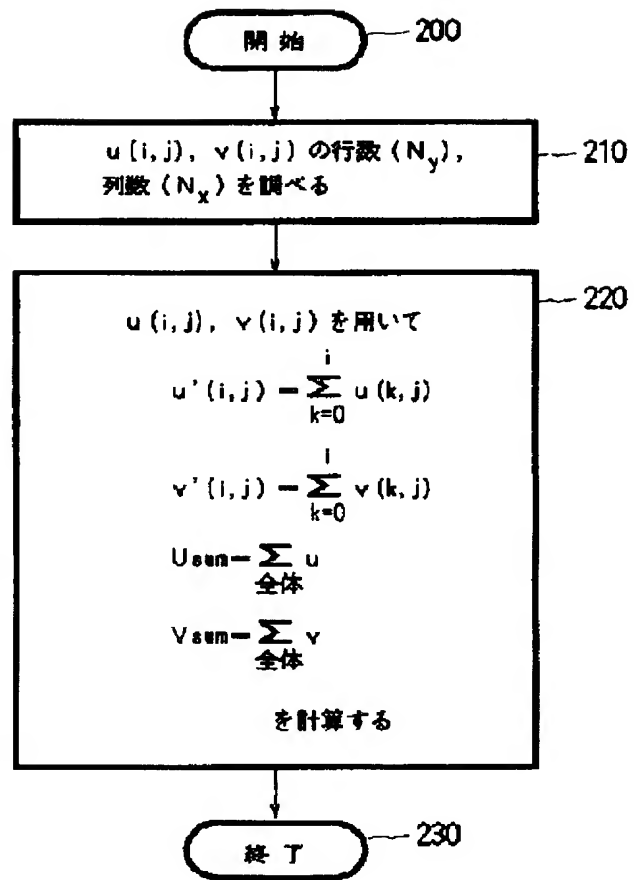
【図6】



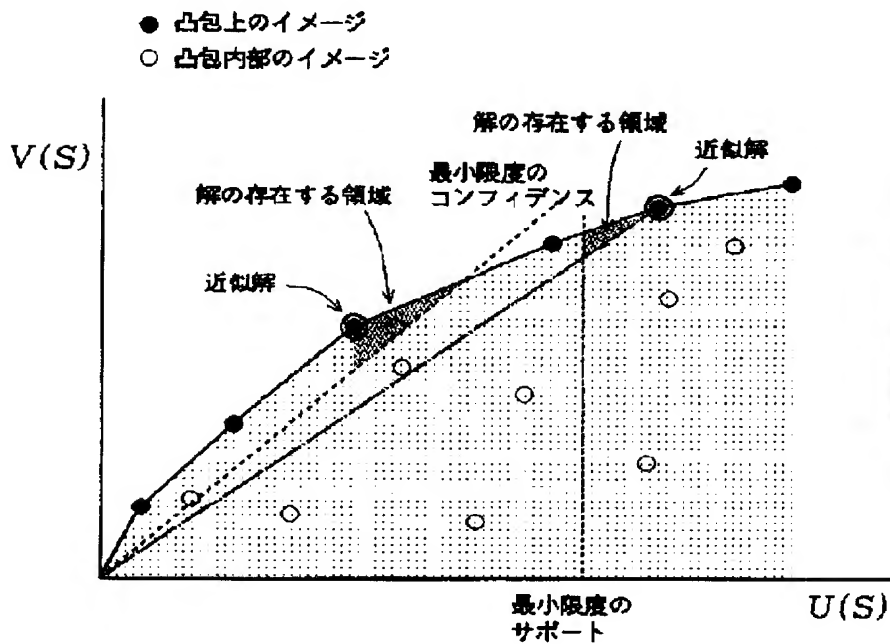
【図1】



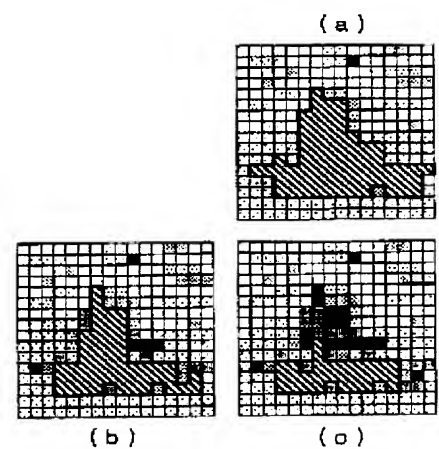
【図2】



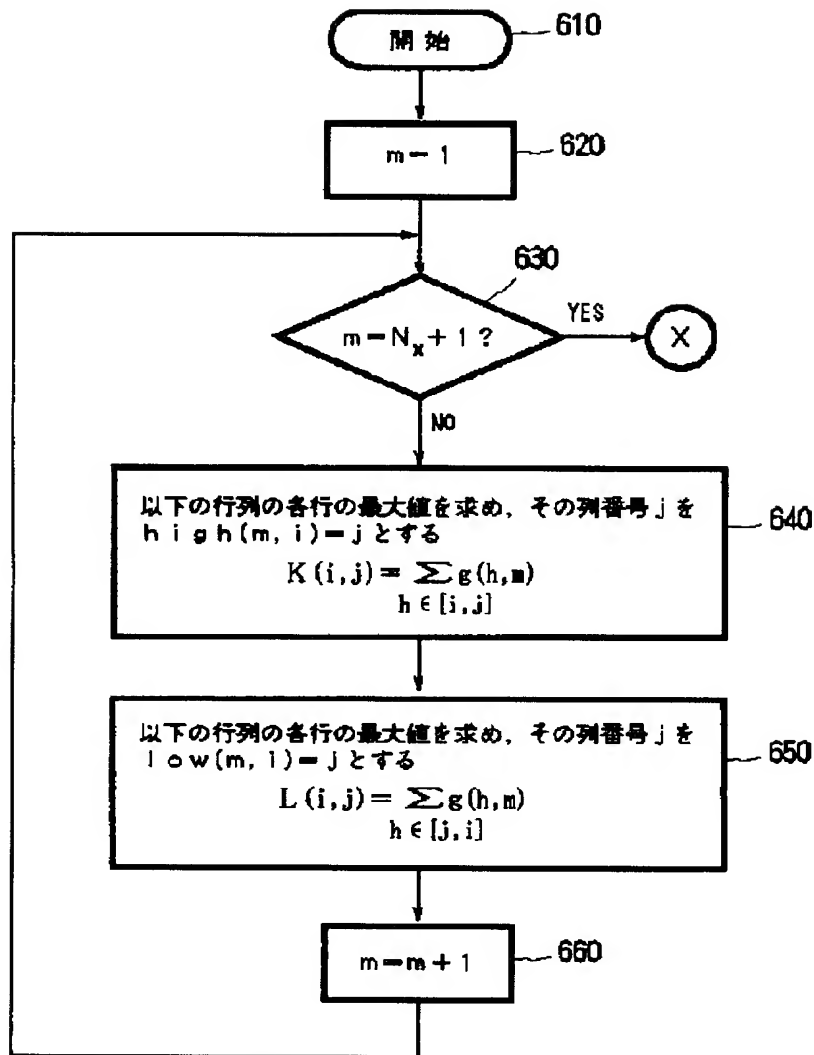
【図5】



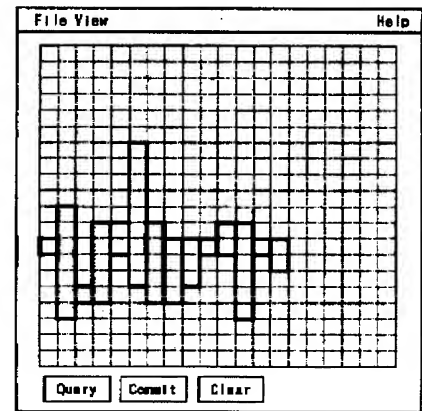
【図10】



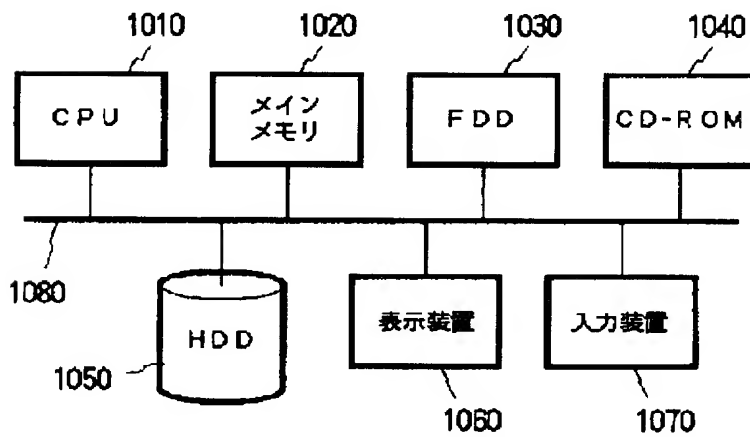
【図7】



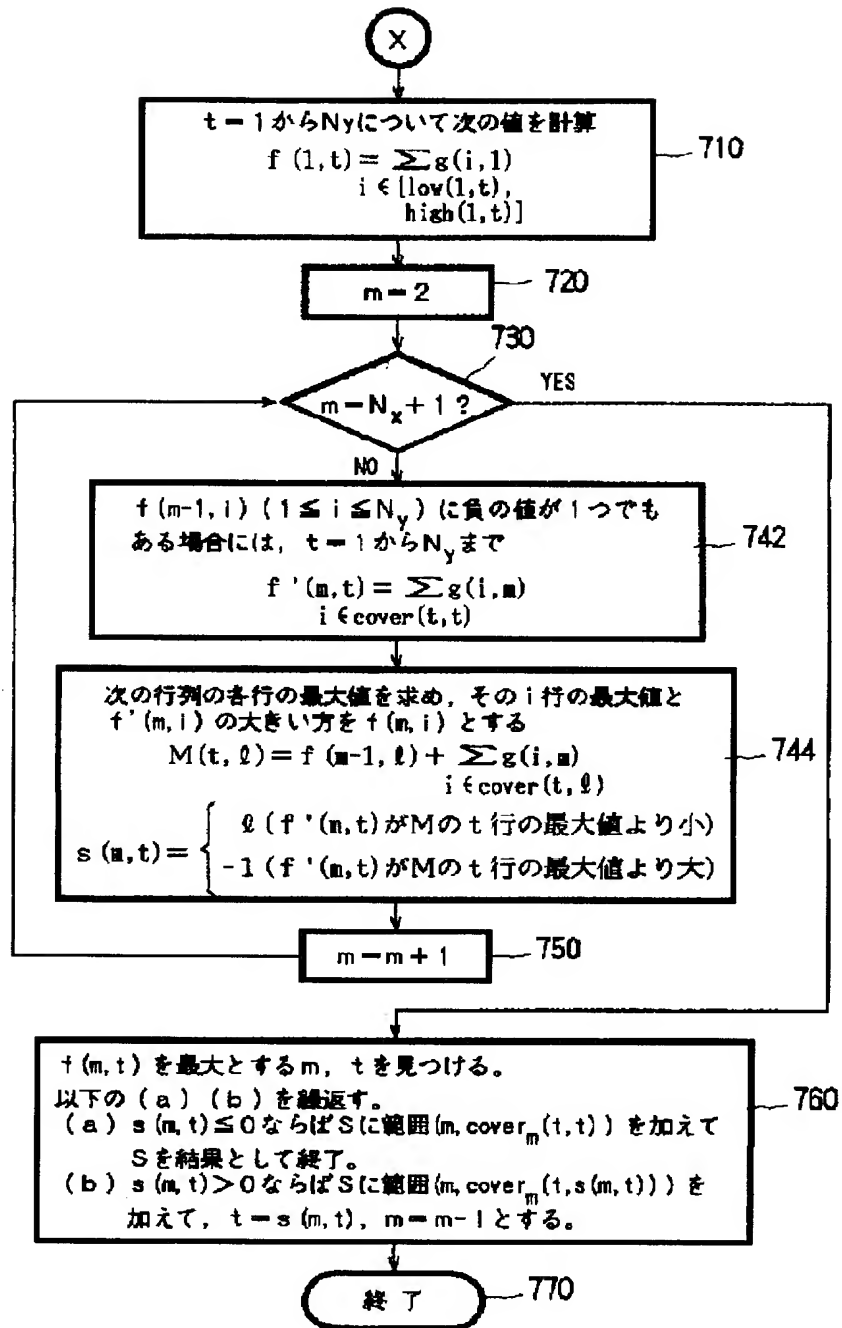
【図17】



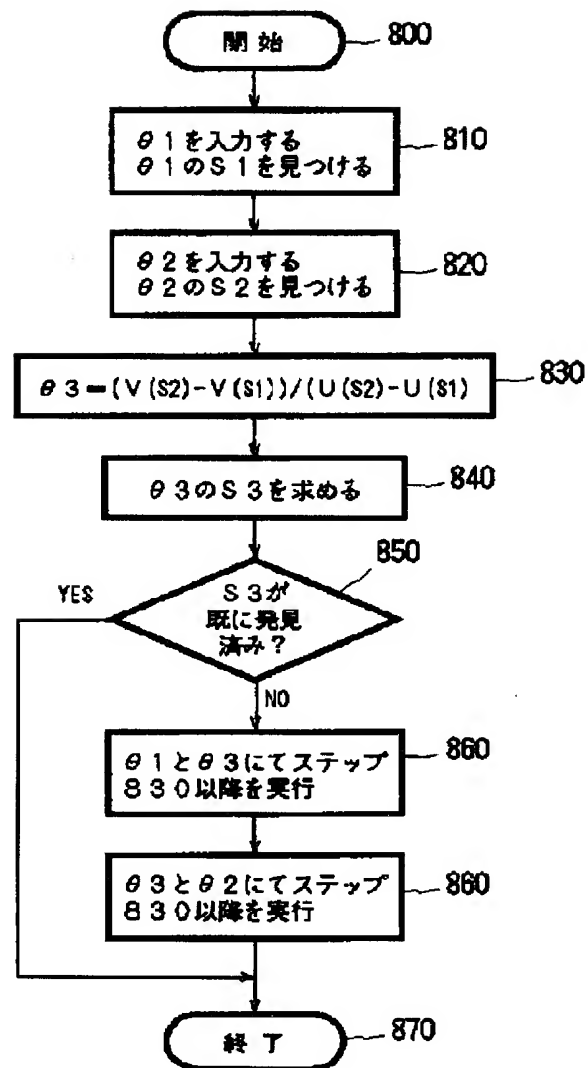
【図15】



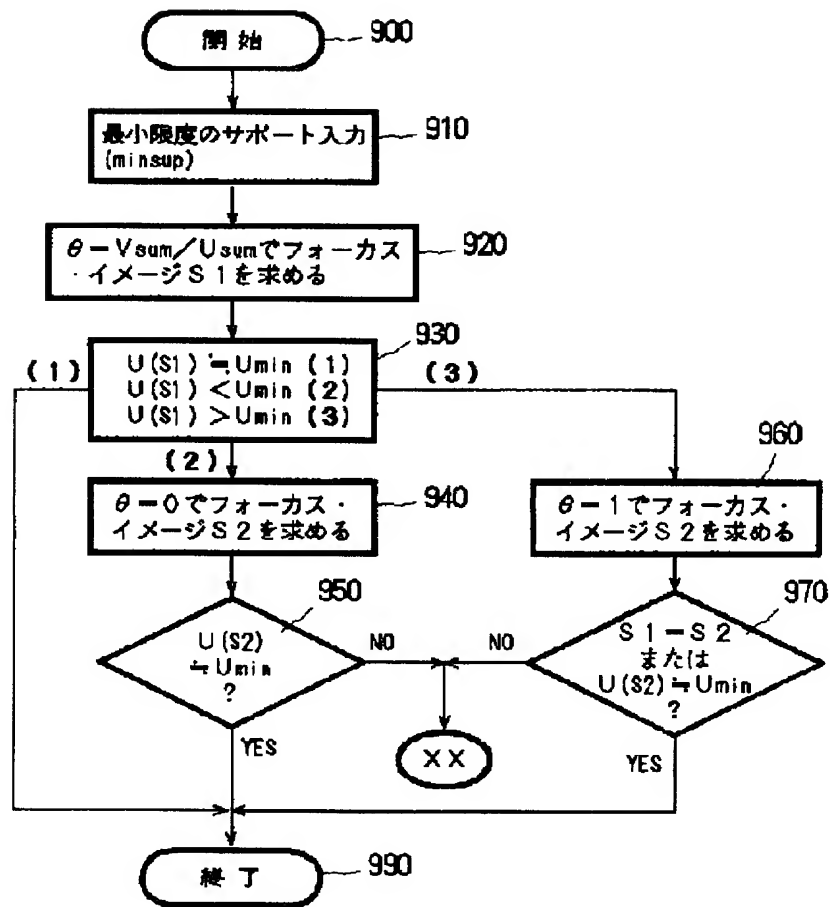
【図8】



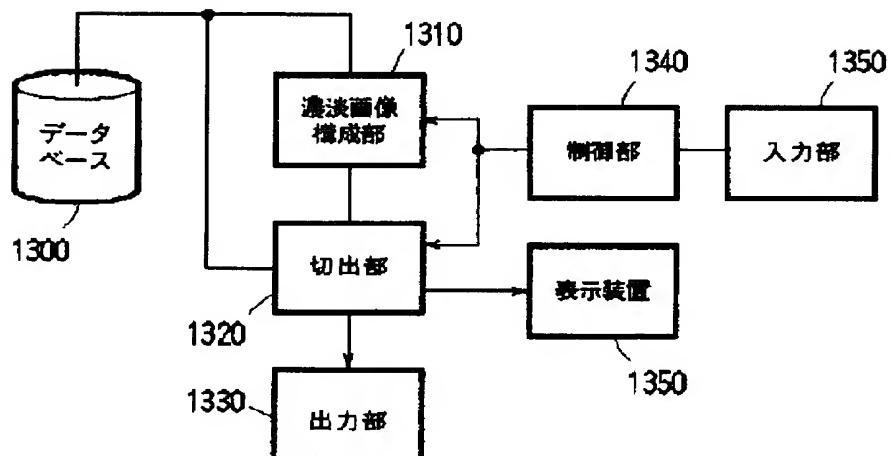
【図9】



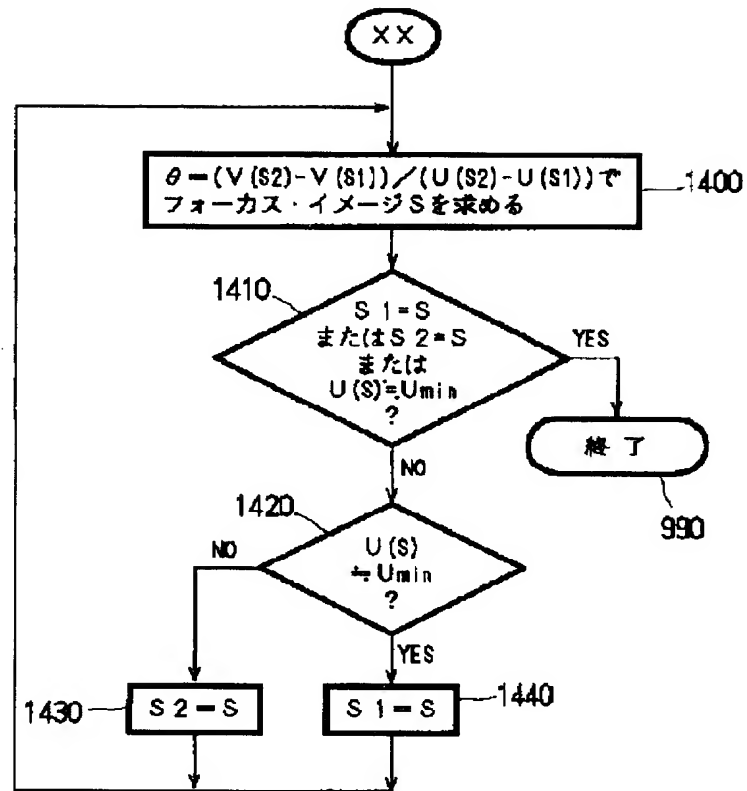
【図11】



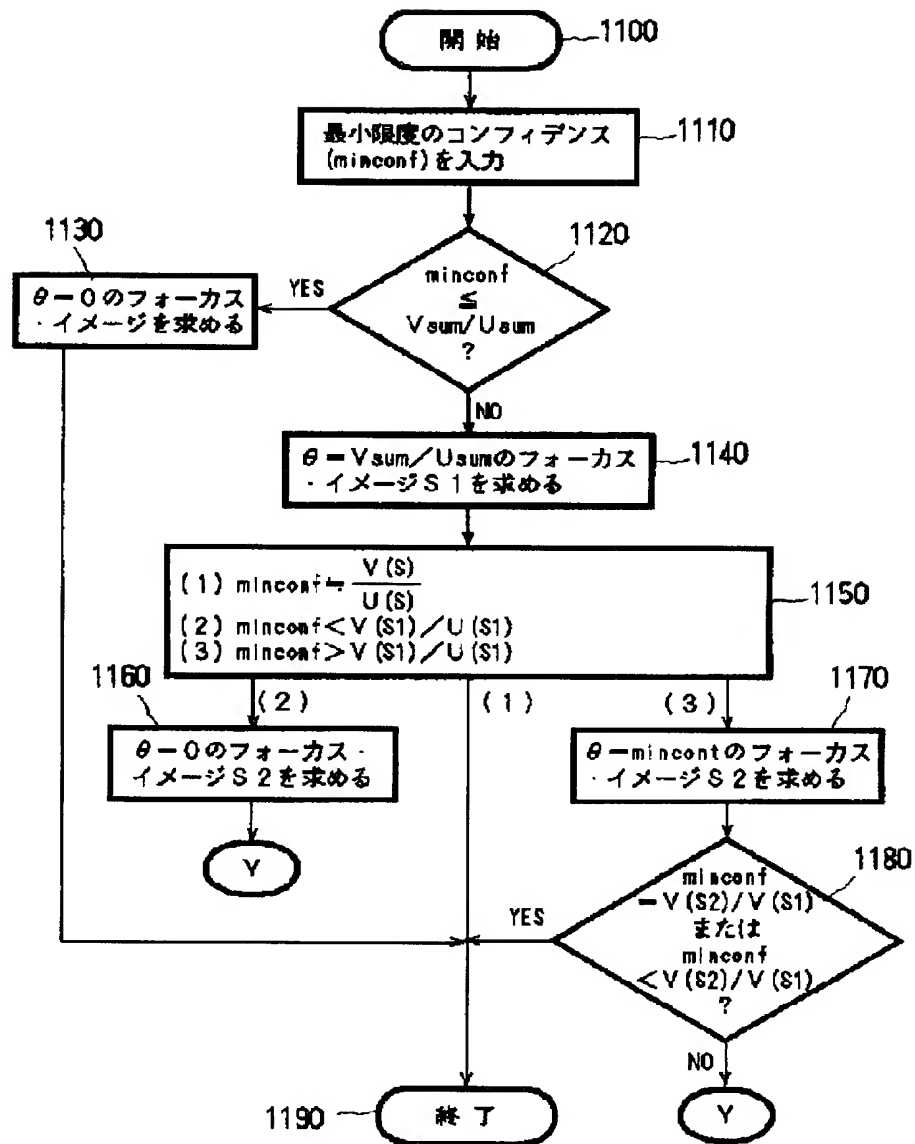
【図16】



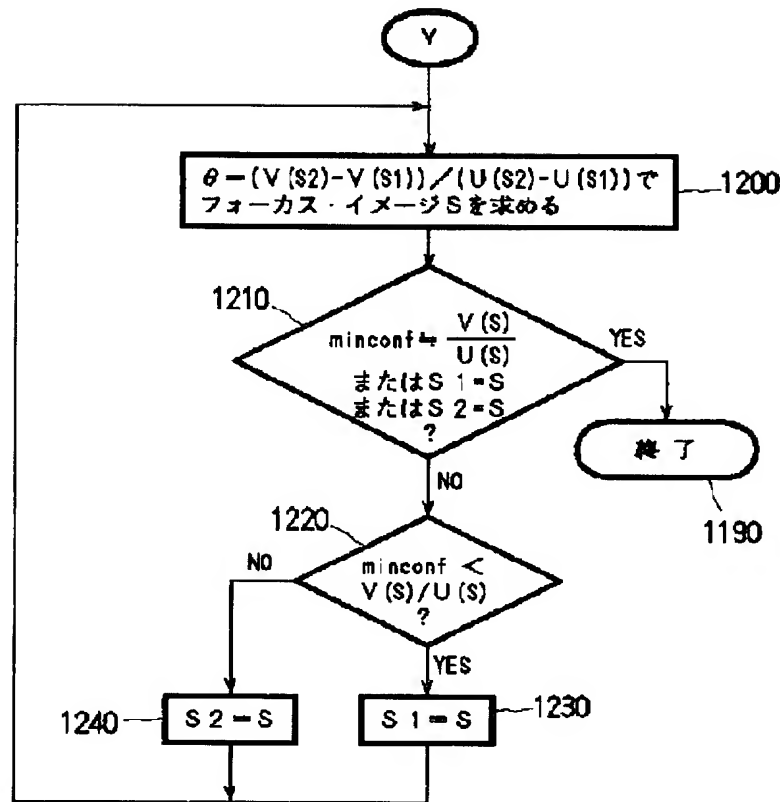
【図12】



【図13】



【図14】



【手続補正書】

【提出日】平成8年6月5日

【手続補正1】

【補正対象書類名】明細書

【補正対象項目名】請求項8

【補正方法】変更

* 【補正内容】

【請求項8】切り出された前記領域Sに対し、
【数7】

$$f(U(S), V(S))$$

$$\begin{aligned}
 &= -\frac{V(S)}{U_{\text{sum}}} \log \frac{V(S)}{U_{\text{sum}}} - \frac{U(S)-V(S)}{U_{\text{sum}}} \log \frac{U(S)-V(S)}{U_{\text{sum}}} \\
 &\quad - \frac{V_{\text{sum}}-V(S)}{U_{\text{sum}}-U(S)} \log \frac{V_{\text{sum}}-V(S)}{U_{\text{sum}}-U(S)} \\
 &\quad - \frac{U_{\text{sum}}-V_{\text{sum}}-U(S)+V(S)}{U_{\text{sum}}-U(S)} \log \frac{U_{\text{sum}}-V_{\text{sum}}-U(S)+V(S)}{U_{\text{sum}}-U(S)}
 \end{aligned}$$

(U_{sum} は前記平面全体のデータ数、 V_{sum} は前記平面全体に含まれる前記真偽をとる属性が真のデータの数を計算し、その値を前記領域Sに対応して記憶するエントロピ計算ステップと、
条件θを変更して前記領域切出ステップと前記エントロピ計算ステップを実行するステップと、
 $f(U(S), V(S))$ を最大化する領域Sを出力するステップとをさらに含む請求項1記載のデータ間結合

ルール導出方法。

【手続補正2】

【補正対象書類名】明細書

【補正対象項目名】請求項15

【補正方法】変更

【補正内容】

【請求項15】切り出されるべき領域に含まれる最低限のデータ数である最小サポート数 U_{min} を入力する手段

と、
 前記切り出された領域Sに含まれるデータ数U(S)と
 前記最小サポート数Uminと比較する手段と、
 前記比較の結果、Umin≧U(S)であれば、当該領域
 Sを切り出されるべき領域として出力する手段と、
 前記比較の結果、Umin>U(S)又はUmin<U(S)
 の場合には、新たな条件θ₃にて、前記領域切出手段が
 動作するように命ずる手段とを含む請求項10記載のデ
 ータ間結合ルール導出装置。

＊【手続補正3】

【補正対象書類名】明細書

【補正対象項目名】請求項17

【補正方法】変更

【補正内容】

【請求項17】切り出された前記領域Sに対し、
 【数12】

$$\begin{aligned} f(U(S), V(S)) &= -\frac{V(S)}{U_{sum}} \log \frac{V(S)}{U_{sum}} - \frac{U(S)-V(S)}{U_{sum}} \log \frac{U(S)-V(S)}{U_{sum}} \\ &\quad - \frac{V_{sum}-V(S)}{U_{sum}-U(S)} \log \frac{V_{sum}-V(S)}{U_{sum}-U(S)} \\ &\quad - \frac{U_{sum}-V_{sum}-U(S)+V(S)}{U_{sum}-U(S)} \log \frac{U_{sum}-V_{sum}-U(S)+V(S)}{U_{sum}-U(S)} \end{aligned}$$

(Usumは前記平面全体のデータ数、Vsumは前記平面全
 体に含まれる前記真偽をとる属性が真のデータの数)を
 計算し、その値を前記領域Sに対応して記憶するエント
 ロピ計算手段と、
 変更された条件θ₃にて前記領域切出手段及び前記エント
 ロピ計算手段が動作するように命ずる手段と、
 前記エントロピ計算手段に記憶されたf(U(S), V
 (S))を最大化する領域Sを出力する手段とを含む請
 求項10記載のデータ間結合ルール導出装置。

【手続補正4】

【補正対象書類名】明細書

【補正対象項目名】請求項21

【補正方法】変更

【補正内容】

【請求項21】コンピュータに、前記切り出された領域
 S内の各バケットのv(i, j)・u(i, j)が、前記平面全体※

$$\begin{aligned} f(U(S), V(S)) &= -\frac{V(S)}{U_{sum}} \log \frac{V(S)}{U_{sum}} - \frac{U(S)-V(S)}{U_{sum}} \log \frac{U(S)-V(S)}{U_{sum}} \\ &\quad - \frac{V_{sum}-V(S)}{U_{sum}-U(S)} \log \frac{V_{sum}-V(S)}{U_{sum}-U(S)} \\ &\quad - \frac{U_{sum}-V_{sum}-U(S)+V(S)}{U_{sum}-U(S)} \log \frac{U_{sum}-V_{sum}-U(S)+V(S)}{U_{sum}-U(S)} \end{aligned}$$

(Usumは平面全体のデータ数、Vsumは平面全体に含ま
 れる真偽をとる属性が真のデータの数)を計算し、その
 値を領域Sに対応して記憶するエントロピ計算ステップ
 と、条件を変更して先の領域切出ステップとエントロピ
 計算ステップを実行するステップと、f(U(S), V
 (S))を最大化する領域Sを出力するステップとを含
 むようにすれば、最適化エントロピ領域を見つけること
 ができる。

【手続補正6】

※のデータ数に対する前記平面全体の前記真偽をとる属性
 のデータ数の割合に等しくなるようv(i, j)を変更させ
 るプログラムコード手段と、
 当該変更されたv(i, j)及び入力された条件θ₃を用い
 て、コンピュータと前記領域切出プログラムコード手段
 が動作するように命じるプログラムコード手段とを有す
 る請求項19記載の記憶装置。

【手続補正5】

【補正対象書類名】明細書

【補正対象項目名】0023

【補正方法】変更

【補正内容】

【0023】ここで、切り出された前記領域Sに対し、
 【数23】

【補正対象書類名】明細書

【補正対象項目名】0074

【補正方法】変更

【補正内容】

【0074】D. 最適化エントロピ・ルールの場合
 最適化エントロピ・ルールとは、領域の内部と外部との
 分割を考えた時、分割前の情報量と比較した分割後の情
 報量の増分を最大化するルールである。よって、切り出
 された領域と平面全体のエントロピのゲイン(以下の

式)が最大となる領域を発見すればよい。

【数42】

$$f(x,y) = -\frac{y}{a} \log \frac{y}{a} - \frac{x-y}{a} \log \frac{x-y}{a} \\ - \frac{b-y}{a-x} \log \frac{b-y}{a-x} \\ - \frac{a-b-x+y}{a-x} \log \frac{a-b-x+y}{a-x}$$

*

*このxはU(S)、yはV(S)、aはUsum、bはVsumである。このような条件においても、解は凸包上に存在することが分かったので、上述のステップを用いることができる。よって、θを変化させ、数42を最大化するフォーカス・イメージを求めればよい。

フロントページの続き

(72)発明者 森本 康彦
神奈川県大和市下鶴間1623番地14 日本アイ・ビー・エム株式会社 東京基礎研究所内

(72)発明者 森下 真一
神奈川県大和市下鶴間1623番地14 日本アイ・ビー・エム株式会社 東京基礎研究所内

(72)発明者 徳山 豪
神奈川県大和市下鶴間1623番地14 日本アイ・ビー・エム株式会社 東京基礎研究所内